

Ministerul Educației al Republicii Moldova
Proiectul *Educație de calitate în mediul rural din Republica Moldova*

Informatica

Ghid de implementare
a curriculumului modernizat
pentru treapta liceală

CARTIER

Elaborat și editat în cadrul Proiectului Educație de calitate
în mediul rural din Republica Moldova, finanțat de Banca Mondială.
Aprobat la ședința Consiliului Național pentru Curriculum, proces-verbal nr.12 din 5 noiembrie 2010.
Aprobat prin Ordinul nr.810 din 9 noiembrie 2010 al ministrului Educației.

Recenzenți:

Vasile Baș, *profesor, grad didactic superior, LT „C. Stere”, Soroca*
Eugen Pîrvan, *profesor de informatică, grad didactic superior, Colegiul Industrial-Pedagogic, Cahul*

CARTIER

Editura Cartier, SRL, str. București, nr. 68, Chișinău, MD2012.
Tel./fax: 24 05 87, tel.: 24 01 95. E-mail: cartier@cartier.md
www.cartier.md

Cărțile CARTIER pot fi procurate în toate librăriile bune din România și Republica Moldova.

LIBRĂRIILE CARTIER

Casa Cărții, bd. Mircea cel Bătrân, nr. 9, Chișinău. Tel./fax: 34 64 61. E-mail: casacartii@cartier.md
Librăria din Centru, bd. Ștefan cel Mare, nr. 126, Chișinău. Tel./fax: 21 42 03. E-mail: librariadincentru@cartier.md
Librăria din Hol, str. București, nr. 68, Chișinău. Tel.: 24 10 00. E-mail: librariadinhol@cartier.md
Librăria 9, str. Pușkin, nr. 9, Chișinău. Tel.: 22 37 83. E-mail: libraria9@cartier.md

Colecția *Cartier Educațional* este coordonată de Liliana Nicolaescu-Onofrei

Editor: Gheorghe Erizanu

Autori: Sergiu Corlat, Lilia Ivanov, Valentin Bîrsan

Lector: Emilian Galaicu-Păun

Coperta: Vitalie Coroban

Design/tehnoeditare: Ana Cioclo

Prepress: Editura Cartier

Tipărită la Tipografia Centrală (nr. 3520)

Sergiu Corlat, Lilia Ivanov, Valentin Bîrsan

**INFORMATICA. GHID DE IMPLEMENTARE A CURRICULUMULUI MODERNIZAT
PENTRU TREAPTA LICEALĂ**
Ediția I, decembrie 2010

© 2010, Ministerul Educației, pentru prezenta ediție.

Toate drepturile rezervate. Cărțile Cartier sînt disponibile în limita stocului și a bunului de difuzare.

Descrierea CIP a Camerei Naționale a Cărții

Informatica. Ghid de implementare a curriculumului modernizat pentru treapta liceală / Sergiu Corlat, Lilia Ivanov, Valentin Bîrsan. – Ch.: Cartier, 2010 (F.E.-P. „Tipogr. Centrală”). – 112 p.: (Colecția „Cartier educațional”).

ISBN 978-9975-79-647-7

37.016.046:004

C 17

Autori:

Sergiu CORLAT,

profesor de informatică, grad didactic superior,
Liceul Teoretic „Orizont”, Chişinău, coordonator

Lilia IVANOV,

consultant,
Agenţia de Evaluare şi Examinare

Valentin BÎRSAN,

profesor, grad didactic superior,
şcoala profesională nr. 5, mun. Bălţi

Consultant ştiinţific:

Ilie LUPU,

profesor universitar, doctor habilitat

1. Structura, funcțiile și modalitățile de aplicare ale curriculumului modernizat (2010)	5
2. Formarea și dezvoltarea personalității în procesul de implementare a curriculumului modernizat	8
3. Metodologia formării competențelor specifice disciplinei	10
4. Corelarea competențelor, subcompetențelor, conținuturilor, tipu- rilor de activități, strategiilor de predare-învățare-evaluare	14
5. Strategii didactice de predare-învățare	20
5.1. Tipologia și specificul strategiilor didactice	20
5.2. Repere și modalități de proiectare a strategiilor didactice	21
5.3. Diversificarea formelor de învățare. Învățarea autonomă. Instruirea la distanță	25
5.4. Realizarea interdisciplinarității	29
5.5. Centrarea pe elev.....	30
5.6. Utilizarea TIC	32
6. Strategii de evaluare	36
6.1. Evaluarea axată pe competențe	36
6.2. Tipuri de evaluare	37
6.3. Metode și tehnici de evaluare	37
6.4. Forme de evaluare curentă	42
6.5. Evaluarea centrată pe succes	44
6.6. Materiale didactice suport pentru evaluarea competențelor	45
7. Recomandări metodice de utilizare a echipamentului și mate- rialelor didactice, a manualelor existente și noi în procesul de implementare a curriculumului modernizat	47
8. Deosebiri ale procesului de predare-învățare-evaluare de profiluri ...	51
9. Recomandări metodice: proiectarea didactică pe profiluri și clase ...	79
10. Proiecte de lecții, centrate pe formarea la elevi a competențelor specifice disciplinei	86
11. Anexe: modele de realizare a instruirii în bază de problemă, în bază de proiect, studii de caz	96
Bibliografie	112

1. Structura, funcțiile și modalitățile de aplicare ale curriculumului modernizat (2010)

Definiții. La fel ca pentru multe alte noțiuni, există diferite definiții ale termenului *curriculum*, care reflectă dezvoltarea teoriilor pedagogice și etapele istorice ale formării acestei noțiuni. Una dintre cele mai relevante este:

În sens larg, prin curriculum se înțelege ansamblul proceselor educative și al experiențelor de învățare prin care trece elevul pe durata parcursului său școlar. În sens restrâns, curriculumul cuprinde ansamblul acelor documente școlare de tip regulator în cadrul cărora se consemnează datele esențiale privind procesele educative și experiențele de învățare pe care școala le oferă elevului. Acest ansamblu de documente poartă, de regulă, denumirea de curriculum formal sau oficial (Al. Crișan, *Curriculum și dezvoltare curriculară: un posibil parcurs strategic*, în Revista de pedagogie, nr. 3-4/1994)

Structura. Curriculumul de bază descrie funcțiile generale ale procesului educațional, sarcinile, scopurile și finalitățile acestuia, stabilește setul de competențe generale, care urmează să fie dezvoltate la cei instruiți. Pe următorul nivel se plasează *curriculumul pe discipline (cursuri)*, care stabilește nișa disciplinei în procesul educațional, sarcinile și scopurile ei specifice în contextul obținerii de către cei instruiți a unor competențe specifice disciplinei date. *Curriculumul pe treaptă de studii* specifică modul de dezvoltare a competențelor proprii disciplinei ținând cont de particularitățile de vîrstă ale elevilor și de cunoștințele acumulate la o anumită treaptă de școlarizare (școală primară, gimnaziu, liceu).

Funcții. Curriculumul de bază are trei categorii de funcții statutare. Realizarea acestora va permite tuturor tinerilor:

- să aibă o instruire orientată pe succes care le va permite
- să-și dezvolte individualitatea personalității capabile să evolueze permanent, să trăiască în condiții de siguranță, viața sănătoasă pentru a deveni în continuare
- cetățeni responsabili, care aduc o contribuție pozitivă pentru dezvoltarea societății.

Aceste obiective statutare trebuie să țină cont de toate aspectele procesului de predare-învățare și să fie punctul de plecare pentru proiectarea curriculumului.

Instruirea orientată pe succes presupune educarea elevilor care

- au abilități esențiale de învățare (limbă și literatură, științe sociale și reale, tehnologia informației și comunicațiilor)
- sînt creativi, capabili să localizeze resurse, să identifice și rezolve probleme din viața reală

- posedă o gândire critică, pot analiza informația, reflecta asupra ei, efectua cercetări, obține rezultate care să și le evalueze
- comunică într-o diversitate de moduri, pot identifica greșelile proprii și au capacitatea de a învăța din ele
- sînt capabili să învețe în mod independent și împreună cu colegii
- cunosc ideile, evenimentele și fenomenele majore care modelează lumea noastră
- se bucură de învățare și sînt motivați pentru a obține cele mai bune rezultate în prezent și în viitor.

Dezvoltarea individualității personalității presupune educarea elevilor care:

- au un simț al valorii proprii și identității personale
- pot comunica și forma relații de calitate în mediul social
- sînt conștienți de sine își pot controla emoțiile
- au valori și idei autentice, posedă principii clare pentru a distinge binele de rău
- progresează în spiritul de independență personală, de inițiativă și de organizare
- optează pentru o viață sănătoasă, se dezvoltă fizic și intelectual
- pot să își asume riscuri și să identifice condiții de siguranță
- își cunosc talentele lor și au ambiții de a le demonstra
- sînt dispuși să încerce lucruri noi și să folosească pe cît e posibil oportunitățile oferite
- sînt deschiși la ideile inspirate de lumea naturală și realizările umane.

Formarea cetățenilor responsabili presupune educarea elevilor care:

- au o pregătire de calitate pentru viață și muncă
- au un spirit întreprinzător
- pot coopera cu alți indivizi din mediul social
- respectă părerile altora, păstrînd integritatea opiniei sale personale
- cunosc propria cultură și tradiție, dar și culturile și tradițiile altor popoare în contextul patrimoniului civilizației umane și au capacitatea de a identifica nișa culturii naționale în acest patrimoniu
- apreciază beneficiile diversității
- nu acceptă manifestările nedreptății, respectă drepturile omului și principiile conviețuirii pașnice
- contribuie la menținerea și îmbunătățirea mediului, la nivel local și global
- țin seama de nevoile generațiilor prezente și viitoare în alegerile pe care le fac
- pot schimba lucrurile în bine

Disciplina Informatica participă la formarea și dezvoltarea generală a personalității, accentual instruirii fiind pus pe dezvoltarea gândirii logice și algoritmice,

pe formarea de competențe digitale. Integrarea persoanei în mediul informatizat al societății moderne este posibilă numai în cazul deținerii cunoștințelor informatice fundamentale și abilităților de utilizare instrumentală și de comunicare cu calculatorul și prin intermediul acestuia – totalitate de competențe care se conțin în noțiunea de cultură informațională.

Curriculumul la disciplina Informatică determină finalitățile didactice generale ale disciplinei, și anume:

- formarea deprinderilor practice de utilizare a calculatorului pentru prelucrarea informației;
- formarea deprinderilor practice de utilizare a rețelor de calculatoare și a serviciilor de rețea;
- formarea deprinderilor practice de comunicare folosind rețelele de calculatoare;
- studierea informaticii ca știință care contribuie la formarea competențelor digitale de bază: elemente de algoritmizare, modelare, programare, gândire logică, acumularea, păstrarea și prelucrarea digitală a informației.

Curriculumul liceal la Informatică propune un model de studiu integrat al acestei discipline, model care contribuie la formarea la elevi a unei concepții unitare asupra informaticii ca știință și asupra metodelor de implementare a conceptelor informatice pentru dezvoltarea perpetuă a societății contemporane.

În acest context, se conturează următoarele **principii** specifice ale disciplinei *Informatică*:

1. **Principiul abordării integrate a disciplinei** – structurarea conținuturilor într-un model integrat, modular, concentric, care are ca scop crearea și dezvoltarea competențelor digitale ale elevului în scopul utilizării sistemelor informatice și cultivarea continuă a modului de gândire algoritmică.
2. **Principiul centrării activității/demersului didactic pe elev** – acceptarea unui model de învățare activă, centrat pe elev, orientat către activități individuale sau în grup, care să permită dezvoltarea independenței de acțiune, a originalității, a creativității, a capacității de lucru în echipă, combinând acestea cu individualizarea ritmului de învățare.
3. **Principiul funcționalității/utilității sociale a procesului didactic** – dezvoltarea aptitudinilor și competențelor necesare pentru integrarea organică a elevilor în societatea informațională. Principiul este realizat în baza rezolvării unor situații de problemă, depășirea cărora contribuie la formarea capacităților de autoperfecționare (autoinstruire).
4. **Principiul corelației interdisciplinare** – abordarea unui demers didactic interdisciplinar cu toate disciplinele școlare, prin utilizarea principiilor și metodelor informatice pentru rezolvarea de probleme, elaborarea proiectelor, prelucrare de informații specifice disciplinelor și utilizarea resurselor educaționale digitale.

2. Formarea și dezvoltarea personalității în procesul de implementare a curriculumului modernizat

Două dintre cele trei grupuri de funcții ale curriculumului țin nemijlocit de formarea și dezvoltarea personalității și anume: *dezvoltarea individualității personalității, formarea cetățenilor responsabili*. Particularizările acestor grupuri pentru disciplina Informatica, în contextul formării competențelor transversale în general și a competenței digitale în special presupune educarea unor cetățeni capabili să se integreze organic în societatea modernă, bazată pe tehnologii.

Dezvoltarea continuă a bazelor științifice ale informaticii duce în consecință la extinderea cantitativă și calitativă a gamei tehnologiilor digitale și, prin inducție, a dispozitivelor digitale. Aceasta impune disciplina școlară informatica să dezvolte competențele ce țin de algoritmizarea situațiilor și problemelor vieții reale pentru o soluționare optimă; modelarea cu ajutorul instrumentelor Informatică a obiectelor, proceselor, fenomenelor, evenimentelor; competența de programare și control ale dispozitivelor digitale complexe. Indiferent de domeniul în care va activa ulterior, absolventul liceului trebuie să poată formula corect secvențe de instrucțiuni, pentru a fi executate, dar să aibă competența de a executa corect diverse operații în baza unor secvențe de instrucțiuni existente.

Dezvoltarea gamei de instrumente digitale pentru prelucrarea informației, pe de o parte, și extinderea capacităților de transfer a datelor, pe de alta, duc la creșterea exponențială a cantității de informație accesibile în orice punct al globului. În acest context disciplina Informatica trebuie să contribuie la crearea competenței tânărului cetățean de a prelucra volume mari de date, de a realiza eficient operații de căutare, selectare, structurare, organizare a informației atât direct, cât și cu ajutorul unor instrumente informatice specializate, în particular aplicațiile de oficiu pentru editarea textelor, calculul tabelar, gestionarea bazelor de date.

Comunicarea în secolul XXI s-a extins din spațiul real în sfera virtuală. Mai întâi poșta electronică, apoi chatul, forumurile, urmate de apariția comunităților virtuale, a rețelelor sociale impun tot mai intens prezența noastră în spațiul digital. În acest context informatica, ca disciplină școlară, are rolul nu numai de a crea la elevi competențe de comunicare digitală, organizare și plasare a informației personale în rețea, dar și să formeze elemente atitudinale față de utilizarea corectă a resurselor hardware, software și comunicaționale, respectarea normelor etice de comunicare în rețea, utilizarea corectă a informațiilor și respectarea legislației virtuale și a drepturilor de autor.

Dezvoltarea perpetuă a tuturor domeniilor de activitate socială implică pentru fiecare membru al societății moderne necesitatea de instruire continuă, pe

parcursul vieții. Competența de „a învăța să înveți” urmează să fie formată încă pînă la absolvirea liceului. Un rol central în formarea acestei competențe îl capătă informatica. Există mai multe argumente, care confirmă această afirmație:

- majoritatea absolută a metodelor de instruire pe parcursul vieții sînt organizate prin instruire mixtă sau la distanță. Aceasta impune prezența la persoana instruită a competențelor de comunicare în spațiul virtual, folosind instrumente digitale pentru comunicare sincronă și asincronă.
- procesul de instruire pe parcursul vieții este realizat de cele mai multe ori independent, prin urmare persoana instruită trebuie să aibă capacitatea de a analiza suportul de curs, urmînd apoi să sintetizeze noi cunoștințe, iar în caz de necesitate – să caute și să selecteze resurse alternative, maximal apropiate cerințelor individuale ale instruitului.
- „spațiul individual de învățare” care conține totalitatea de instrumente virtuale, resurse, aplicații, canale de comunicare, etc. necesare celui instruit este organizat în baza unor obiecte web. Competența de a crea și a dirija aceste obiecte este creată inițial de disciplina informatică.
- Metodele moderne de instruire: învățarea colaborativă, în bază de problemă, în bază de proiect, portofoliu electronic etc., nu ar putea fi eficient realizate fără a conține componenta digitală.
- Instruirea interactivă, sistemele de testare, laboratoarele virtuale și aplicațiile de simulare impun utilizatorilor săi prezența unei gândiri algoritmice, capacității de a abstractiza, de a utiliza modele de diferite tipuri, de a formula exact instrucțiuni, de a acționa cu precizie. Toate acestea inițial se formează în cadrul disciplinei informatica.

Astfel, formarea competențelor specifice disciplinei Informatica duce în consecință la educarea unor personalități care se pot încadra în societatea informațională, tehnologică modernă; pot comunica calitativ folosind cele mai diverse medii de comunicare; au capacitatea de a învăța continuu; folosesc competența de a gândi algoritmic în viața cotidiană, prin urmare – contribuie la dezvoltarea continuă a întregii societăți, fiind în același timp conștienți de pericolele majore ale dezvoltării tehnologice necontrolate.

3. Metodologia formării competențelor specifice disciplinei

Pînă a stabili metodologia formării competențelor specifice disciplinei Informatica vom defini noțiunea de competență școlară, din ea vom separa caracteristicile unei competențe, iar avînd caracteristicile vom determina metodologia de formare a competenței. Procesul este unul general, comun pentru toate disciplinele școlare, iar metodologia diferă doar în funcție de conținuturile didactice și setul de resurse educaționale specifice disciplinei.

Conform curriculumului național competența școlară este definită ca „*un ansamblu/sistem integrat de cunoștințe, capacități, deprinderi și atitudini dobîndite de elevi prin învățare și mobilizate în contexte specifice de realizare, adaptate vârstei elevului și nivelului cognitiv al acestuia, în vederea rezolvării unor probleme cu care acesta se poate confrunta în viața reală.*”

O definiție mai puțin formală poate stabili competența ca fiind *capacitatea de a rezolva problemele din viața cotidiană utilizînd în timp real cunoștințele și atitudinile dobîndite.*

Caracteristicile unei competențe

Vom analiza acum competența propriu-zisă. Ea este definită¹ prin cinci caracteristici esențiale:

Mobilizarea unui ansamblu de resurse. Competența face apel la mobilizarea unui ansamblu de resurse: cunoștințe, experiențe, scheme, automatisme etc. Doar această caracteristică nu este suficientă pentru a face deosebirea între capacitate și competență.

Caracterul finalizat. A doua caracteristică a competenței constă în faptul că mobilizarea ansamblului de resurse nu se face în mod întîmplător. Competența este finalizată: resursele sînt folosite conștient de purtătorul competenței, ele capătă o „funcționalitate”.

Relația cu un ansamblu de situații. Cea de-a treia caracteristică ține de faptul că mobilizarea resurselor are loc în cadrul unui ansamblu de situații reale și în contextul rezolvării unor probleme reale.

Caracterul disciplinar. Este în strînsă legătură cu cea precedentă. În timp ce capacitățile au un caracter transversal, competențele au un caracter disciplinar. Această caracteristică rezultă din faptul că competența este deseori definită printr-o categorie de situații corespunzătoare unor probleme specifice disciplinei,

¹ Xavier Roegiers. *Manualul școlar și formarea competențelor în învățămînt.*

provenite și ele direct din exigențele disciplinei. Fără îndoială, anumite competențe ce aparțin diferitelor discipline sînt uneori apropiate una de alta, fiind, așadar, mai ușor transferabile.

De exemplu, competența de a „efectua experimente virtuale, rezolva probleme de activitate cotidiană și elabora modele ale fenomenelor studiate, folosind aplicații, laboratoare și medii digitale educaționale; interpretarea rezultatelor obținute” specifică disciplinei Informatica este apropiată de competențele de cercetare asociate științelor exacte din cursul liceal.

Evaluabilitatea. Competența este o mărime evaluabilă, deoarece poate fi măsurată prin calitatea îndeplinirii sarcinii și prin calitatea rezultatului.

Etapale formării competențelor în procesul de studiu

Formarea unei competențe școlare este un proces extins în timp, separabil în etape convențional distincte (*fuzzy*), cu elemente de trecere comune. Tradițional aceste etape țin de:

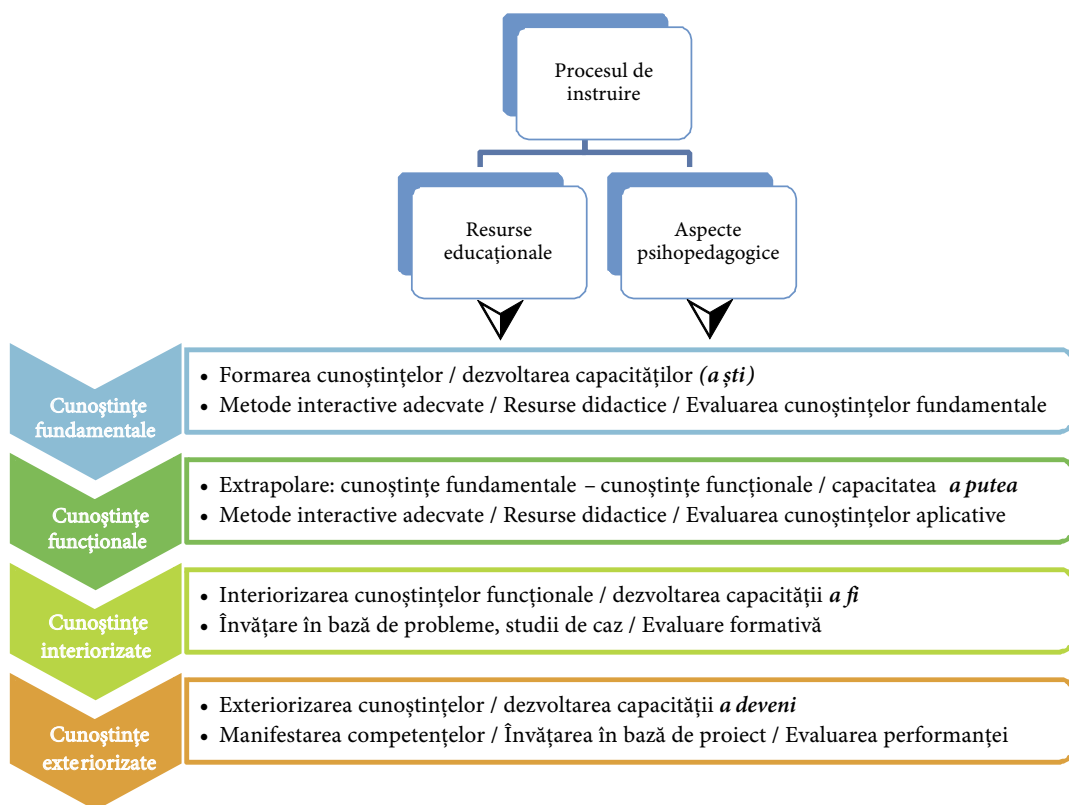
- achiziționarea cunoștințelor fundamentale (știu)
- transformarea cunoștințelor fundamentale în cunoștințe funcționale (știu să fac)
- interiorizarea cunoștințelor/formarea atitudinilor (știu să fiu)
- exteriorizarea cunoștințelor (știu să devin)/această etapă se referă tradițional la competențele științifice care, în contextul curriculumului modernizat, devin o parte a activității școlare.

Etapale de formare a competenței școlare se condiționează reciproc și constituie un ciclu continuu care, prin utilizarea diferitor metode, forme, mijloace de învățare, are drept rezultat avansarea performanței.

În proiectarea conținutului educațional orientat la formarea competențelor urmează să se țină cont de următoarele condiții:

- respectarea sistemului de principii specifice cunoașterii științifice, ținînd cont de particularitățile de vîrstă;
- organizarea procesului educațional în baza teoriilor cunoașterii științifice, dezvoltării cognitive (L. Vîgotsky), învățării experimentale (D. Kolb);
- realizarea etapelor de formare a competențelor: *cunoștințe fundamentale, funcționale, interiorizate, exteriorizate.*

Etaple și condițiile stabilite anterior permit elaborarea modelului metodologic de formare a competențelor (competențe specifice, subcompetențe), fig. 1:



Nivelul cunoștințelor fundamentale. Bazat pe receptare, memorare, atenție. Pune bazele formării și dezvoltării individului. Cunoștințele se bazează prioritar pe memorare. Informațiile se memorează fără penetrarea esenței problemei. Instruiții pot defini noțiuni, fapte, legi; reproduc unele date.

Nivelul cunoștințelor funcționale. Permite instruitului să aplice cunoștințele însușite. Se utilizează preponderant perceperea, sesizarea prin simțuri, se caută algoritmi de operare cu informații, se inițiază capacitățile mintale de: *analiză, sinteză, generalizare*.

Nivelul cunoștințelor interiorizate. Dezvoltă procesele mintale de analiză, sinteză, generalizare. Pentru antrenarea atitudinilor se folosesc pe larg rezolvarea situațiilor-problemă, idealizările, modelările.

Nivelul de performanță. Este nivelul în care instruitul operează singur cu unele cunoștințe interiorizate, exteriorizându-le în rezolvarea situațiilor semnificative (care depășesc problemele cotidiene), demonstrând prin aceasta performanțe individuale.

În particular, pentru disciplina Informatica, în baza reperelor expuse anterior, pot fi propuse următoarele sugestii metodologice:

Fiind un ansamblu de capacități, cunoștințe și abilități, competența poate fi realizată doar prin dezvoltarea integrată a aspectelor sale dominante:

- *aspectul cognitiv*, care vizează utilizarea teoriilor și a noțiunilor din Informatică;
- *aspectul funcțional*, care reprezintă capacitățile persoanei de a activa într-un anumit domeniu: profesional, educațional, social, utilizând mijloacele digitale;
- *aspectul etic*, care vizează valorile personale și sociale.

Aspectul cognitiv, de asimilare a informației comunicate, implică procesele psihice de percepție, memorare și operații de gândire. Prin urmare, pentru elaborarea sarcinilor didactice se va utiliza în special taxonomia lui Bloom, orientată spre atingerea de către persoana instruită a nivelului intelectual, determinat de standarde.

Pentru asimilarea de către elevi a cunoștințelor se recomandă utilizarea metodelor: SINELG, interviu, lectura ghidată, exerciții practice la calculator, probleme simple pentru dezvoltarea gândirii algoritmice;

Aspectul de expertiză (funcțional) al competenței are rolul de dezvoltare a capacităților intelectuale și psihomotorii ale elevilor. Pentru dezvoltarea potențialului intelectual pot fi folosite taxonomiile: Simpson, Dove etc.

Metodele recomandate la disciplina Informatica sînt: expunerea de material teoretic, lucrul la calculator, individual și/sau sub conducerea cadrului didactic, rezolvarea de probleme, lucrarea practică, lucrarea de laborator.

Aspectul aplicativ (integrator, atitudinal) al competenței formează la elevi atitudini și comportament în contextul condițiilor sociale bine determinate. Pentru atingerea acestui scop, la elaborarea sarcinilor didactice se va folosi taxonomia lui Krathwohl.

Metodele recomandate în acest context sînt studiul de caz, proiectul de cercetare, dezbaterile etc.

4. Corelarea competențelor, subcompetențelor, conținuturilor, tipurilor de activități, strategiilor de predare-învățare-evaluare

Corelarea: competențe transversale – specifice – subcompetențe

COMPETENȚE-CHEIE/TRANSVERSALE, TRANSDISCIPLINARE: au fost definite pe baza competențelor-cheie descrise de Comisia Europeană. Cu unele excepții, acestea sînt formulate în același mod în toate programele. Din ele, au fost derivate COMPETENȚELE SPECIFICE, care reprezintă detalieri pentru fiecare obiect de studiu al celor 10 competențe-cheie, pentru treapta liceală de învățămînt. SUB-COMPETENȚELE sînt etape ale competențelor specifice, pe trepte/ani de studiu.

Corelarea corectă a competențelor de diferite nivele (transversal, specific, sub-competență) asigură realizarea integrală a funcțiilor curriculumului de bază.

Pentru disciplina Informatica, corelarea competențelor la treapta liceală este determinată de tabelul 1.

În context vom folosi următoarele notații:

CT(X) – competența transversală cu indicele X CS (X) – competența specifică cu indicele X.

Indicii corespund celor din următoarele liste:

Competențe transversale:

- CT1. Competențe de învățare/de a învăța să înveți.
- CT2. Competențe de comunicare în limba maternă/limba de stat.
- CT3. Competențe de comunicare într-o limbă străină.
- CT4. Competențe acțional-strategice.
- CT5. Competențe de autocunoaștere și autorealizare.
- CT6. Competențe interpersonale, civice, morale.
- CT7. Competențe de bază în matematică, științe și tehnologie.
- CT8. Competențe digitale, în domeniul tehnologiei informației și comunicațiilor (TIC);
- CT9. Competențe culturale, interculturale (de a recepta și a crea valori).
- CT10. Competențe antreprenoriale.

Competențe specifice:

- CS1. Formarea unei viziuni științifice asupra componentei informatice în societatea contemporană.
- CS2. Cunoașterea proceselor, principiilor și metodelor de codificare și decodificare a informației în scopul realizării comunicării interumane și umane – sistem informatic.

- CS3. Identificarea structurii generale a sistemelor digitale, a principiilor de funcționare a sistemelor de transmitere, stocare și de prelucrare a informației.
- CS4. Elaborarea modelelor informatice ale obiectelor, sistemelor și proceselor frecvent întâlnite în activitatea cotidiană.
- CS5. Aplicarea metodelor de algoritmizare, de formalizare, de analiză, de sinteză și de programare pentru soluționarea problemelor legate de prelucrarea automatizată a informației.
- CS6. Translarea algoritmilor frecvent utilizați într-un limbaj de programare de nivel înalt.
- CS7. Colectarea, păstrarea și prelucrarea informației cu ajutorul aplicațiilor software specializate.
- CS8. Crearea și elaborarea documentelor Web.
- CS9. Efectuarea experimentelor virtuale, rezolvarea problemelor de activitate cotidiană și elaborarea de modele ale fenomenelor studiate, folosind aplicații, laboratoare și medii digitale educaționale; interpretarea rezultatelor obținute.
- CS10. Folosirea competențelor informatice pentru căutarea și selectarea informațiilor în interes de autoinstruire și orientare profesională.
- CS11. Respectarea dreptului de autor asupra resurselor digitale, a normelor de etică și securitate informațională. Protejarea de infracțiunile informatice.

Elemente de formare și dezvoltare ale competențelor transversale CT2, CT3, CT4, CT5, CT6, CT9 se regăsesc în toate competențele specifice ale disciplinei, precum și în subcompetențe, dar nu poartă un caracter dominant. Din acest motiv în tabelul de corelare sînt incluse doar competențele transversale și specifice dominante pentru fiecare compartiment al curriculumului. Cu (*) sînt marcate subcompetențele și compartimentele asociate doar profilului real.

Tabelul 1. Corelare: competențe transversale/specifice/subcompetențe

Competențe transversale	Competențe specifice	Subcompetențe
Clasa X		
1. TIPURI DE DATE STRUCTURATE		
CT1 CT7 CT8	CS1 CS5 CS6	• Argumentarea necesității structurării datelor. • Recunoașterea și utilizarea formulelor metalingvistice și a diagramelor sintactice ale declarațiilor de tipuri de date în studiu. • Prelucrarea datelor structurate. • Alegerea structurii de date, adecvate rezolvării unei probleme.

2. INFORMAȚIA		
CT1 CT7 CT8	CS1 CS2 CS5 CS6	- Identificarea conceptelor de bază pentru reprezentarea și transmiterea informației. - Utilizarea mijloacelor informatice în activitatea cotidiană. - Estimarea cantității de informație transmisă, recepționată și prelucrată în activitățile cotidiene. - Aplicarea principiilor de bază de codificare și decodificare a informației pentru transmiterea, recepționarea și prelucrarea ei.
3. BAZELE ARITMETICE ALE TEHNICII DE CALCUL		
CT1 CT7 CT8	CS1 CS2 CS5 CS6	- Identificarea limitelor de reprezentare a numerelor în calculator. - Aplicarea aritmeticii de calculator pentru soluționarea problemelor de prelucrare a informației. * Aplicarea operațiilor de bază ale aritmeticii de calculator la soluționarea problemelor informatice.
4. * ALGEBRA BOOLEANĂ		
CT1 CT7 CT8	CS1 CS2 CS5	- Utilizarea algebrei booleene pentru prelucrarea digitală a informației. - Argumentarea necesităților de utilizare a algebrei booleene în informatică.
5. * CIRCUITE LOGICE		
CT1 CT7 CT8	CS1 CS2 CS3	- Reprezentarea variabilelor logice prin mărimi fizice. - Utilizarea proiectării ierarhice la elaborarea structurilor de calcul, prelucrare și stocare a informațiilor.
6. STRUCTURA CALCULATORULUI ȘI REȚELE DE CALCULATOARE		
CT1 CT7 CT8 CT10	CS1 CS2 CS3 CS7 CS8 CS11	- Utilizarea resurselor tehnice și a resurselor programate ale calculatorului. - Utilizarea dispozitivelor externe de memorare pentru stocarea curentă și de lungă durată a informațiilor. - Utilizarea dispozitivelor de intrare-ieșire ale calculatorului. - Clasificarea rețelelor de calculatoare. - Utilizarea tehnologiilor de cooperare în rețea. - Descrierea principiilor de organizare și utilizarea serviciilor Internet.
Clasa XI		
1. SUBPROGRAME		
CT1 CT7 CT8	CS1 CS3 CS4 CS5 CS6	- Prelucrarea datelor cu ajutorul subprogramelor predefinite și a subprogramelor elaborate de către utilizator. - Organizarea comunicării între programul/subprogramul apelant și subprogramul apelat. - Proiectarea structurală a algoritmului și a programului. * Utilizarea recursiei pentru rezolvarea problemelor.

2. * STRUCTURI DINAMICE DE DATE		
CT1 CT7 CT8	CS1 CS3 CS4 CS5 CS6	- Identificarea problemelor a căror soluționare necesită utilizarea structurilor dinamice de date. - Utilizarea structurilor dinamice de date pentru soluționarea problemelor întâlnite în activitatea cotidiană. - Implementarea și gestionarea structurilor dinamice de date frecvent utilizate. - Gestionarea memoriei interne a calculatorului.
3. * TEHNICI DE PROGRAMARE		
CT1 CT7 CT8	CS1 CS4 CS5 CS6	- Estimarea complexității algoritmilor. - Utilizarea tehnicilor de programare la rezolvarea problemelor din diferite domenii. - Alegerea tehnicii de programare adecvate problemei.
Clasa XII		
1. * ELEMENTE DE MODELARE		
CT1 CT7 CT8 CT10	CS1 CS4 CS5 CS6 CS9	- aplicarea criteriilor de clasificare a modelelor; - elaborarea modelelor matematice; - motivarea importanței modelării în activitatea economică și viața socială; - identificarea soluțiilor analitice și soluțiilor de simulare; - selectarea tipului soluției în funcție de natura problemei; - planificarea și realizarea procesului de rezolvare a unei probleme la calculator;
2. * CALCUL NUMERIC		
CT1 CT7 CT8 CT10	CS1 CS4 CS5 CS6 CS7 CS9 CS10	- identificarea valorilor exacte și a aproximărilor acestora; - determinarea erorii absolute și a erorii relative; - evaluarea erorilor de calcul, generate de erorile datelor de intrare; - estimarea erorilor, generate de particularitățile reprezentării numerelor în calculator;
		- utilizarea algoritmilor elementari pentru separarea soluțiilor pe un interval dat; - identificarea condițiilor de aplicare a metodei bisecției (coardelor, Newton); - elaborarea într-un limbaj de programare de nivel înalt a programelor de calcul iterativ al soluției ecuației algebrice sau transcendente prin metoda bisecției (coardelor, Newton); - alegerea metodei de rezolvare a ecuațiilor algebrice și transcendente (bisecției, coardelor, Newton) adecvate pentru o problemă dată;
		- selectarea tehnicii de implementare a algoritmului; - elaborarea subprogramelor pentru calculul numeric al determinanților; - selectarea tehnicii de implementare a algoritmului; - elaborarea, într-un limbaj de programare de nivel înalt, a subprogramelor pentru rezolvarea sistemelor de ecuații liniare;

		• elaborarea programelor (subprogramelor) pentru calculul numeric al integralelor prin metoda dreptunghiurilor în funcție de un număr de divizări, stabilit apriori; • identificarea problemelor, rezolvarea cărora se reduce la calculul unei integrale definite;
3. BAZE DE DATE		
CT1 CT7 CT8 CT9 CT10	CS1 CS4 CS5 CS7 CS10 CS11	• descrierea structurii bazelor de date ierarhice, în rețea și relaționale; • descrierea structurii și a funcțiilor sistemelor de gestiune a bazelor de date; • distingerea etapelor de elaborare a unei baze de date; • cunoașterea rolului persoanelor antrenate în elaborarea și utilizarea bazelor de date; • proiectarea structurii tabelelor bazei de date; • crearea tabelelor cu ajutorul sistemului de asistență sau prin proiectare independentă; • utilizarea metodelor de introducere a datelor în tabele; • crearea și editarea tabelelor pentru problemele frecvent întâlnite în matematică, fizică, biologie, chimie, geografie etc.; • stabilirea corelațiilor între tabele; • utilizarea operațiilor destinate sortării înregistrărilor, căutării și înlocuirii valorilor; • elaborarea filtrelor pentru selectarea înregistrărilor;
		• alegerea tipurilor de interogări adecvate pentru prelucrarea datelor; • elaborarea interogărilor cu ajutorul sistemului de asistență; • elaborarea interogărilor pentru selectarea datelor;
		• identificarea componentelor unei expresii și aplicarea regulilor de formare a expresiilor; • elaborarea interogărilor de acțiune; • utilizarea tehnicilor de grupare și totalizare a datelor;
		• identificarea elementelor care alcătuiesc formularele; • elaborarea formularelor cu ajutorul sistemului de asistență; • aplicarea tehnicilor de modificare a formularelor; • utilizarea formularelor pentru vizualizarea, modificarea și validarea datelor; • elaborarea formularelor pe baza tabelelor corelate;
		• identificarea componentelor care alcătuiesc raportul; • elaborarea rapoartelor și subrapoartelor cu ajutorul sistemului de asistență; • utilizarea tehnicilor de modificare a rapoartelor; • utilizarea tehnicilor de grupare a datelor într-un raport;
4. ELEMENTE DE WEB DESIGN		
CT1 CT7 CT8 CT9 CT10	CS1 CS7 CS8 CS10 CS11	• identificarea cerințelor și recomandărilor pentru documente Web; • cunoașterea etapelor de elaborare a unui document Web; • identificarea documentelor Web în format HTML; • elaborarea documentelor HTML cu ajutorul aplicațiilor de oficiu; • publicarea documentelor Web în Internet;

	• elaborarea documentelor HTML simple; • formatarea textului cu ajutorul instrumentelor HTML; • crearea și organizarea listelor cu ajutorul instrumentelor HTML; • crearea și utilizarea legăturilor interne și externe; • inserarea în documente HTML a imaginilor; • crearea și editarea tabelor în documente HTML; • utilizarea tabelor pentru amplasarea în pagină a elementelor HTML.
--	--

Corelarea: subcompetențe – conținuturi didactice este stabilită nemijlocit în curriculumul liceal la disciplină. De asemenea, corelarea re apare în procesul elaborării planificării de lungă durată, care poate fi structurată tradițional sau pe unități de învățare.

Corelarea: conținuturi didactice – activități – strategii didactice

Ultimul nivel de corelare este stabilit în procesul elaborării proiectelor didactice ale lecțiilor sau ale unităților de conținut.

5. Strategii didactice de predare-învățare

5.1. Tipologia și specificul strategiilor didactice

Strategia didactică este un termen unificator, care reunește sarcinile și situațiile de învățare, reprezentînd un sistem integru de mijloace, metode și resurse educaționale ce vizează dezvoltarea unor competențe. Strategia didactică ocupă un loc central în activitatea didactică, deoarece proiectarea și organizarea lecției se realizează în funcție de decizia strategică a profesorului. Ea este concepută ca un scenariu didactic complex, în care sînt implicați actorii predării-învățării, condițiile realizării și metodele vizate. Prin urmare, strategia stabilește traseul metodic optim pentru abordarea unei situații concrete de predare/învățare. În acest fel, prin proiectare strategică se pot preveni erorile, riscurile și evenimentele nedorite din activitatea didactică.

În calitate de elemente factice, metodele sînt cosubstanțiale strategiilor. Cu alte cuvinte, strategia nu se confundă cu metoda sau metodologia didactică, deoarece acestea din urmă vizează o activitate de predare-învățare-evaluare, în timp ce strategia vizează procesul de instruire în ansamblu.

Principalele componente ale strategiei didactice sînt:

- sistemul formelor de organizare și desfășurare a activității educaționale
- sistemul metodologic (metodele și procedeele didactice)
- sistemul mijloacelor de învățămînt (resurse utilizate)
- sistemul obiectivelor operaționale/subcompetențelor (pentru o lecție/unitate de învățare).

Dintre strategiile didactice mai importante, aplicabile la studiul disciplinei Informatica menționăm:

- *strategii inductive* (demersul didactic: de la particular spre general);
- *strategii deductive* (demersul didactic: de la general spre particular);
- *strategii analogice* (învățarea în baza modelelor);
- *strategii mixte* (inductiv-deductiv și deductiv-inductiv);
- *strategii algoritmice* (demers demonstrativ, intuitiv, structurat, ordonat);
- *strategii euristice* (obținerea cunoștințelor prin efort propriu – problematizare, experimentare, analiză, sinteză).

De cele mai multe ori specialiștii din domeniul informaticii folosesc strategiile algoritmice, mixte și analogice, îmbinîndu-le cu elemente de dirijare și independență, cu accent pe predare-învățare semidirijată. Pentru obținerea de către elevi a performanțelor în domeniul informaticii recomandăm utilizarea strategiilor euristice, cu orientare către autoinstruirea ghidată.

Strategiile didactice sînt realizate cu ajutorul *metodelor* de predare și învățare, informative și activ-participative, de studiu individual, de verificare și evaluare.

5.2. Repere și modalități de proiectare a strategiilor didactice

Componentele care determină o strategie didactică sunt: *formele* activităților educaționale, *metodele* de învățare folosite în cadrul activităților, *mijloacele* folosite de profesor/elevi pe parcursul activităților și *finalitățile urmărite* de profesor/elevi. Prin urmare, pentru proiectarea unei strategii didactice este necesară selectarea formelor, metodelor, mijloacelor și finalităților adecvate și corelarea lor corectă pe întreg parcursul realizării strategiei.

Formele de activități didactice

Formele de activități didactice se clasifică după diverse criterii, principalele fiind:

- durata activității: *an școlar, semestru, sistem de lecții (unitate de învățare), lecție, moment de lecție;*
- grupul țintă: *clasă, grup, elev individual;*
- modul de organizare: *formal* (cu flux informațional formativ sistematic, omogen, programat, cuantificat și dirijat însoțit de o evaluare formativă continuă și sistematică pentru atingerea finalităților educaționale și dezvoltarea psihică a elevilor; *nonformal/neformal* (activități școlare și extrașcolare unde influențele educative sînt structurate, organizate și instituționalizate în afara sistemului tradițional de învățare fiind variate, flexibile, oferind posibilitatea de a pune în evidență interesele, abilitățile și opțiunile particulare ale elevului), *informal* (fluxul informațional este aleatoriu, neorganizat, nedorit, nesistematic, cunoștințele sau abilitățile dobîndite sînt aleatorii, neselectate, căpătate involuntar, ca urmare a trăirii unor experiențe de viață).

Astfel, forma activității didactice stabilește cadrul temporal, auditorial și organizatoric în care se integrează metodele utilizate pentru transmiterea demersului didactic și mijloacele de învățare folosite în timpul activității didactice.

Metode de învățare

Metode de învățare – căile prin care elevii ajung, în procesul de învățămînt, sub coordonarea educatorilor, la dobîndirea de cunoștințe, deprinderi, la dezvoltarea capacităților intelectuale și la valorificarea aptitudinilor specifice.

Metoda este un ansamblu de operații mintale și practice, prin care elevul dezvoltă esența evenimentelor, proceselor, fenomenelor, cu ajutorul profesorului sau independent.

În sens larg, metoda este o practică raționalizată, o generalizare confirmată de experiența curentă sau de experimentul psihopedagogic și care servește la transformarea și ameliorarea naturii umane.

Principalele funcții ale metodelor de învățare sunt: *cognitivă* (dirijarea cunoașterii pentru a obține noi cunoștințe); *normativă* (modul de predare-învățare); *motivațională* (de stimulare a interesului pentru învățare); *formativ-educativ-compensatorie* (exersare, antrenare și dezvoltare a proceselor psihice).

Principalele metode de învățare sunt: *expunerea și conversația didactică, demonstrația, observarea, lucrul cu manualul, exercițiul, algoritimizarea, modelarea didactică, instruirea în bază de problemă, instruirea programată, studiul de caz, simularea, învățarea prin descoperire, proiectul de cercetare*. Majoritatea metodelor sînt bine cunoscute și descrise în literatura de specialitate. Ne vom opri la descrierea doar a unor metode, care devin esențiale în contextul realizării unui curriculum axat pe competențe pentru disciplina Informatica.

Instruirea în bază de problemă

Este denumită și predare prin rezolvare de probleme.

O definiție de lucru a metodei este: metoda didactică ce constă din punerea în fața elevului a unor dificultăți create în mod deliberat, în depășirea cărora, prin efort propriu, elevul învață ceva nou. Poate fi folosit sistemul de clasificare a situațiilor de problemă după Kudreavțev:

- 1) există un dezacord între vechile cunoștințe ale elevului și cerințele impuse de rezolvarea unei noi situații;
- 2) elevul trebuie să aleagă dintr-un lanț sau sistem de cunoștințe, chiar incomplete, numai pe cele necesare în rezolvarea unei situații date, urmînd să completeze datele necunoscute;
- 3) elevul este pus în fața unei contradicții între modul de rezolvare posibil din punct de vedere teoretic și dificultatea de aplicare a lui în practică;
- 4) elevul este solicitat să sesizeze dinamica mișcării într-o schemă aparent statică;
- 5) elevului i se cere să aplice, în condiții noi, cunoștințele anterior asimilate.

Necesitatea utilizării acestei metode în învățămînt este ușor de dovedit, date fiind numeroasele ei valențe convenabile educației actuale: faptul că favorizează aspectul formativ al învățămîntului, prin participarea efectivă și susținută a elevului și prin dezvoltarea intereselor de cunoaștere; faptul că sporește aplicabilitatea informației elevului în practică; faptul de a crea elevului o posibilitate de transfer a diverselor reguli însușite etc.

Exemplu: Anexa 1a: Problema „Zone de risc”

Instruirea în bază de proiect

Instruirea în bază de proiect este o metodă de instruire/evaluare care permite elevilor să capete cunoștințe semnificative și abilități practice necesare pentru o carieră de succes. Dacă ne orientăm serios către atingerea obiectivelor educaționale principale ale secolului XXI, atunci această metodă trebuie să fie în centrul atenției.

O instruire riguroasă și profundă în bază de proiect este organizată în baza unei întrebări sau situații principale deschise. Astfel elevii se axează pe aprofundarea învățării prin centrarea asupra temelor importante, dezbateri, întrebări și/sau rezolvări de probleme.

Metoda creează necesitatea de a cunoaște conținuturile esențiale și a căpăta competențele specifice. Proiectele tipice (și cele mai multe instruirii) încep prin prezentarea către elevi a cunoștințelor și conceptelor, iar odată cu asimilarea lor oferă posibilitatea aplicării practice. Realizarea proiectului începe cu prezentarea scopului final, atingerea căruia necesită cunoștințe specifice și concepte, creînd astfel premise pentru a învăța.

Caracteristicile instruirii în bază de proiect:

- 1) Elevii se angajează în rezolvarea problemelor complexe din lumea reală; dacă este posibil, selectează sau definesc probleme semnificative pentru ei.
- 2) Elevii se implică în activități de cercetare, creează abilități de planificare, dezvoltă gândirea critică, și de rezolvare a problemelor – creează competențe pentru finalizarea proiectului.
- 3) Creează aptitudini/standarde specifice și cunoștințe în contextul activităților realizate în proiect.
- 4) Facilitează învățarea și aplicarea abilităților de comunicare interpersonală, lucrul în echipă.
- 5) Permite elevilor utilizarea practică a unei game de competențe necesare pentru dezvoltarea profesională și carieră (planificarea timpului/resurselor; responsabilitatea individuală etc.).
- 6) Include perspective de atingere a finalităților de studii (referite la curriculum) stabilite la începutul proiectului.
- 7) Încorporează activități de reflecție ale studenților pentru analiza critică a experiențelor în cadrul proiectului și de corelare a experiențelor la standardele specifice de învățare.
- 8) Finalizează cu o prezentare sau un produs care să demonstreze realizarea sarcinilor de învățare.

Exemplu: Anexa 1b: Proiect „Analiza eficienței algoritmilor de sortare”

Studiul de caz

Definiția de lucru: metodă ce constă din confruntarea elevului cu o situație reală de viață, prin a cărei observare, înțelegere, interpretare, urmează să realizeze un progres în cunoaștere.

Pentru un studiu de caz clasic se pot identifica următoarele etape:

- a) alegerea cazului și conturarea principalelor elemente semnificative;
- b) lansarea cazului, care poate avea loc în mai multe forme, între care lansarea ca o situație problematică;
- c) procurarea informației în legătura cu cazul;
- d) sistematizarea materialului, prin recurgera la diverse metode, între care cele statistice;
- e) dezbaterea asupra informației culese, care poate avea loc prin diverse metode;
- f) stabilirea concluziilor și valorificarea proprie: o comunicare, ipoteze de verificat, decizii.

Toate aceste elemente și dimensiuni ne îndreptățesc să o considerăm o metodă compozită, concentrând în sine o suită întreagă de alte metode, fără de care nu poate exista.

Pentru studiile de caz la disciplina Informatica, în special la primele experiențe ale elevilor, se recomandă realizarea etapelor a) – c) de către profesor, iar la următoarele etape se impune o asistență consistentă pentru membrii echipei care realizează studiul de caz.

Example: Anexa 1c: Studiul de caz „Publicarea documentelor Web”

Anexa 1d: Studiul de caz „Algoritmi de calcul al determinanților”

Mijloace de învățare

Mijloacele de învățare reprezintă ansamblul de resurse (obiecte, instrumente, produse, aparate, echipamente și sisteme tehnice) care susțin și facilitează activitățile de învățare și evaluare în cadrul procesului de predare-învățare.

Clasificarea resurselor educaționale se face în funcție de analizatorul solicitat, precum și de caracterul celor prezentate. Astfel resursele pot fi clasificate în:

- resurse vizuale (texte, prezentări, imagini și dispozitivele de vizualizare ale acestora);
- resurse auditive (înregistrări sonore și dispozitivele de audiere);
- resurse audio-vizuale (înregistrări video cu bandă sonoră și dispozitivele de vizualizare și audiere).

Pentru activitățile de învățare la disciplina Informatica suportul material al resurselor de învățare este calculatorul și echipamentele periferice specializate,

dar nu se exclude și utilizarea unor mijloace de învățare tradiționale (fișe, placate, modele materiale, reportofon, camere video etc.).

Utilizarea corectă, echilibrată a mijloacelor de învățare permite să se evidențieze următoarele caracteristici ale procesului educațional:

- Flexibilitatea** – posibilitatea de a selecta din mulțimea de resurse și dispozitive cele mai adecvate situației la moment.
- Generalitatea** – posibilitatea de a codifica în diferite forme mesajele transmise, astfel încât să fie antrenate câteva sisteme de recepție concomitent.
- Paralelismul** – posibilitatea utilizării simultane a aceleiași resurse în mai multe scopuri sau de către mai mulți utilizatori concomitent.

Specificul disciplinei Informatica impune utilizarea masivă a calculatorului electronic în calitate de mijloc de învățare. Fiind un instrument universal, el poate înlocui, în funcție de dispozitivele periferice conectate și ansamblul de resurse educaționale și aplicații software, toate mijloacele tehnice tradiționale. Posibilitatea de comunicare prin intermediul rețelei Internet cu resursele informaționale externe extinde și mai mult gama metodelor de utilizare a calculatorului în scopuri educaționale.

Totuși, trebuie să ținem cont că un mijloc de învățare, chiar și calculatorul, nu este eficient în sine, ci numai ca element component al unui sistem de forme, metode, mijloace și resurse, care se completează și se integrează într-o *strategie* de instruire sau autoinstruire coerentă.

5.3. Diversificarea formelor de învățare.

Învățarea autonomă. Instruirea la distanță

Pentru învățământul preuniversitar această formă de instruire poate fi aplicată doar în cazuri excepționale, vizându-i elevii care din diverse motive nu pot frecventa instituția de învățământ perioade extinse de timp (an școlar, treaptă de școlarizare).

Instruirea la distanță presupune distanțarea geografică a profesorului și elevului pe parcursul întregii perioade de instruire. Resursele pentru învățare sînt formate din resursele tradiționale (manuale, problemare, materiale didactice) și resurse digitale de diverse tipuri, care pot să se afle atît pe calculatorul elevului instruit, cît și online. Comunicarea în activitățile de învățare este organizată prin sesiuni sincrone și asincrone, asigurate de instrumente software comode și simple în utilizare: chat (text/sunet/video), forumuri, bloguri. Pentru socializarea elevului pot fi organizate și micro conferințe web video sau sonore. Evaluarea poate fi efectuată online pe sisteme de testare distanțate, sau local, pe calculatorul elevului, după care testul este trimis prin email evaluatorului.

O altă situație în care apar elemente de instruire la distanță este *învățarea colaborativă*, care presupune formarea unui grup de profesori și elevi, distanțați geografic, care rezolvă în comun diverse probleme, sau elaborează anumite proiecte informatice. Un element specific al învățării colaborative este necesitatea accesului comun la produsul digital creat, ceea ce în condițiile actuale poate fi realizat fie prin un blog, fie printr-un site sau document cu acces comun.

Instruirea mixtă

În sens larg, instruirea mixtă este definită în felul următor:

Instruirea mixtă este o combinație de metode de învățare tradiționale și online, în scopul maximizării eficienței procesului de instruire. Programul de instruire este divizat în module. Pentru fiecare modul este stabilită aparte cea mai potrivită metodă de livrare către clasă în general și către fiecare elev în parte. Scopul instruirii mixte este de a profita de cele mai bune componente ale instruirii tradiționale și a instruirii la distanță, evitând în același timp neajunsurile. Pot fi utilizate mai multe medii de predare, pornind de la ateliere de lucru tradiționale, predarea în sala de clasă, cărți și materiale tipărite sau alte tipuri de suporturi tradiționale, precum și materiale în bază digitală, resurse distribuite Web, pe purtători digitali, și elemente de e-learning.

<http://dictionary.bnet.com/definition/blended+learning.html>

În învățământul liceal procesul de instruire mixtă poate fi folosit pentru a facilita progresul elevului și a îmbunătăți elementul comunicativ între profesor și elevi în afara orelor de curs. În acest sens este mai potrivit modelul nuclear al instruirii mixte. Planul cadru nu prevede realizarea distanțată a activităților didactice la disciplinele școlare. Prin urmare, elementele de instruire mixtă pot să apară doar în afara orelor de curs. Aceasta implică următoarele tipuri de activități:

- a. **consultații** la tema curentă (sau în general la disciplină). Profesorul stabilește un interval de timp (1 – 2 ore) pe parcursul săptămânii în timpul căruia se află online la un calculator conectat la rețea și răspunde instant, folosind o aplicație de mesagerie (Google talk, Yahoo messenger etc.) la întrebările elevilor, care țin de însușirea cunoștințelor sau de îndeplinirea temei pentru acasă. Pentru elevi de asemenea este necesar accesul la calculatoare conectate la rețea. O variantă asincronă a consultațiilor online poate fi realizată cu ajutorul serviciului de email, al forumurilor tematice sau al blogurilor. În această variantă comunicarea cu elevii poate fi individualizată în sensul alegerii celor mai potrivite intervale de timp pentru comunicare atât pentru elevi, cât și pentru profesor.

- b. **instruirea de performanță.** Pregătirea elevilor pentru concursurile de programare sau proiecte informatice presupune un contact între elevi și profesor nu doar în timpul orelor de curs (cu atât mai mult că aceste ore sînt destinate activităților cu întregul auditoriu), ci și în special în afara lor. Modul de interacțiune rămîne același ca și în cazul precedent, dar grupul de actori și resurse se modifică: pentru comunicare se folosesc instrumentele deja menționate, suplinite cu resurse ale elevilor (probleme, proiecte etc. realizate sau în curs de realizare) plasate într-un mediu virtual (site, blog); în discuția problemelor/proiectelor se pot implica și alți profesori sau colegi, la invitația autorilor, sau din proprie inițiativă. În procesul educațional se adaugă resursele digitale educaționale externe pentru problema/proiectul curent amplasate pe portaluri sau site-uri tematice.
- c. **instruirea elevilor cu deficiențe fizice.** Pentru unii elevi prezența permanentă la orele de curs nu este posibilă din diverse motive. Dacă starea lor fizică permite alternarea perioadelor de absență cu sesiuni de prezență la lecții, eficiența învățării acestora poate fi sporită folosind instruirea mixtă. Forma de comunicare în sesiunile distanțate poate fi aleasă dintre cele enumerate anterior, în funcție de situația concretă: ședințele sincrone au proprietatea de a stimula capacitatea de autoorganizare, dar nu se poate insista asupra lor în cazul unor situații medicale. Și în acest caz profesorul trebuie să dispună de seturi extinse de resurse, care să permită un progres optim al elevului.
- d. **evaluarea online.** Activitatea poate fi inițiată de profesor pentru o evaluare formativă, sau de către elev(i) pentru a stabili nivelul curent sau progresul într-o perioadă de timp. Pentru evaluările inițiate de profesor elevii trebuie să cunoască intervalul sau intervalele de timp în care urmează să susțină testul online, tematica testului, modul de apreciere. Profesorul va lansa activitățile de evaluare online individualizat, pentru elevi sau grupuri, care au posibilitatea de a accesa resursele online în afara orelor de curs.

Utilizarea elementelor de instruire mixtă în liceu permite și trecerea individualizată a elevilor la un alt model de instruire și anume la autoinstruirea ghidată (învățarea autonomă).

Învățarea autonomă

Principiile schemei „nucleare” pot fi folosite și pentru organizarea procesului de autoinstruire ghidată, la orice disciplină. Activitățile independente de învățare ale elevului pot fi generate de cîteva situații tipice:

- a. **Necesitatea de a recupera activității educaționale.** În calitate de resurse de învățare poate servi manualul și materialele didactice tradiționale. Dar setul de resurse pot fi extins considerabil prin resurse plasate online. Profesorul care folosește un asemenea model de instruire trebuie să dispună de o gamă întreagă de resurse, dintre care să le poată propune pe cele care se potrivesc cel mai mult personalității elevului care va învăța independent. Suplimentar, în momentele critice de învățare, elevul care studiază independent poate să apeleze la profesor pentru depășirea acestora. Instrumentele de comunicare rămân aceleași ca și pentru instruirea mixtă: email, chat (text/sunet/video), forum, blog. Aprecierea succesului învățării poate fi determinată de profesor pe parcurs, urmărind progresul elevului prin comunicarea curentă, prin test online la final de recuperare, prin test tradițional sau în orice alt mod comod pentru profesor și elev.
- b. **Autoinstruirea de performanță.** Dacă particularitățile psihice/de vîrstă/de performanță ale elevului permit învățarea autonomă, profesorul poate dirija procesul de învățare stabilind împreună cu elevul finalitățile (competențele) care se doresc a fi obținute, direcțiile progresului, setul minim de resurse și punctele de reper/evaluare pentru stabilirea succesului. În particular, pentru disciplina informatică, organizarea acestui model de învățare este foarte eficientă din motivul prezenței unui număr considerabil de resurse educaționale online.

Autoinstruirea, în special cea orientată spre performanță, presupune capacitatea dezvoltată a elevului (și în general a celui instruit, indiferent de vîrstă) de a gândi critic, de analiză și sinteză, autoevaluare. Pentru a iniția procesul de autoinstruire este necesară o etapă de analiză prealabilă a domeniului (problemei), cînd elevul își stabilește răspunsurile la următoarele întrebări:

- Care sînt competențele pe care vreau să le capăt?
- Ce resurse pot folosi?
- Cine mă poate ajuta?
- Ce activități (acțiuni)-cheie îmi pot ajuta?
- Ce mă poate opri?
- Cu ce să încep?
- Cum să evaluez succesul învățării?

Răspunsurile determină atît resursele de învățare, cît și persoanele care pot să apară în calitate de tutori sau consultanți pe tot parcursul învățării. Contactele stabilite, împreună cu resursele web identificate și persoanele din grupul de interes formează *mediul pentru autoinstruire*.

5.4. Realizarea interdisciplinarității

Interdisciplinaritate: Transfer de concepte și metodologie dintr-o disciplină în alta pentru a permite abordarea mai adecvată a problemelor cercetate. (DEX online)

Promovarea interdisciplinarității constituie un element definitoriu al progresului cunoașterii. Problema interdisciplinarității a preocupat filosofii și pedagogii încă din cele mai vechi timpuri: sofistii greci, Plinius, Comenius și Leibnitz etc.

Interdisciplinaritatea implică un anumit grad de integrare între diferitele domenii ale cunoașterii și între diferite abordări, ca și utilizarea unui limbaj comun permițând schimburi de ordin conceptual și metodologic. Ea prezintă o formă de cooperare între discipline științifice diferite, care se realizează în principal respectând logica științelor în cauză, adaptate particularităților legii didactice, și-l ajută pe elev în formarea unei imagini unitare a realității, îi dezvoltă o gândire integratoare.

În procesul de învățământ se regăsesc demersuri interdisciplinare la nivelul corelațiilor minimale obligatorii, sugerate chiar de curriculumul la disciplină sau pe arii curriculare. Corelarea cunoștințelor și abilităților de la diferitele discipline de învățământ contribuie substanțial la realizarea educației elevilor, la formarea și dezvoltarea flexibilității gândirii, a capacității lor de a aplica cunoștințele în practică.

Legătura dintre discipline se poate realiza la nivel de conținuturi, tehnologii, competențe, dar efectul major este crearea unui mediu integrat, care permite exprimarea liberă a personalității elevului, dezvoltarea capacităților de lucru în echipă, autoinstruire și autoevaluare.

Pentru disciplina informatică transferul interdisciplinar se realizează nu doar la nivel de conținuturi și tehnologii, dar și la nivel de instrumente și resurse. Instrumentele educaționale software, resursele digitale elaborate de echipe de profesori și elevi pot fi folosite cu succes în studierea altor discipline, dar au la bază un aparat informatic fundamental.

Informatica este una dintre științele cu multiple aspecte aplicative. Acestea se reflectă și în structura cursului liceal de informatică. Elementele de interdisciplinaritate apar chiar de la studierea conceptelor informatice fundamentale, care își au originile în logică, matematică, fizică. Deoarece un aspect fundamental al informaticii este prelucrarea informației, tehnologiile și mediile informatice se utilizează din ce în ce mai mult în studierea disciplinelor din sfera umanistică: limbi moderne, geografie, istorie etc., unde eficiența cercetării este proporțională cu capacitatea de prelucrare a informației.

Utilizarea platformelor de eLearning, de management al învățării, sau chiar a resurselor digitale online sau locale presupune prezența la elevi (și profesori) a

unor componente ale competenței digitale care se dezvoltă preponderent în cadrul disciplinei Informatica. Pe de altă parte, utilizarea resurselor educaționale digitale la orele altor discipline au în calitate de rezultat secundar și extinderea prin exersare a competențelor digitale.

Cea mai eficientă metodă de realizare a interdisciplinarității în cadrul disciplinei Informatica este abordarea ei în bază de proiecte aplicative cu echipe mixte (2 – 3 elevi, profesorul la disciplina de profil, profesorul de informatică). Spre deosebire de microproiectele informatice (durata 1 – 2 săptămîni), durata proiectelor interdisciplinare poate fi stabilită, în funcție de complexitatea proiectului, la un semestru sau un an de studii.

Exemple:

- crearea, publicarea și menținerea unui site cu o tematică dată (observații de fenomene, evenimente; sondaje online și analiza lor etc.);
- elaborarea și publicarea ziarelor, revistelor școlare pe platforme digitale;
- elaborarea proiectelor informatice de modelare a fenomenelor fizice, experiențelor și reacțiilor chimice;
- proiectele matematice pentru aproximarea funcțiilor, trasarea graficelor, realizarea calculelor complexe, calculul simbolic etc.

O bună practică este participarea la concursurile de proiecte informatice sau resurse educaționale, cum ar fi, de exemplu: „Infomatrix” (concurs internațional de proiecte informatice: www.infomatrix.ro), Cupa SIVECO (concurs de soft educațional, România), iTineret (concurs de proiecte informatice cu categoriile: site-uri web, aplicații, grafică: Ministerul Tehnologiilor Informaționale și Comunicațiilor, Republica Moldova: www.itineret.md), în cadrul cărora se dezvoltă și competențele de comunicare în diverse situații, limbi și medii.

5.5. Centrarea pe elev

Învățarea centrată pe elev este un model de instruire care plasează elevul în centrul procesului educațional. În acest model elevii sînt participanți activi ai procesului de învățare. Ei pot studia din locații proprii și folosind strategii proprii. Învățarea este individualizată, nu standardizată. Învățarea centrată pe elev dezvoltă asemenea aptitudini de a învăța să învețe cum sunt: rezolvările de probleme, gîndirea critică și reflectivă, competențele de cercetare. Modelul este flexibil și se adaptează ușor la diverse stiluri de învățare ale elevilor.

(National Center for Research on Teacher Learning. 1999)

Trecerea la metodologia activă, centrată pe elev, implică elevul în procesul de învățare pentru a-i crea atît aptitudini de învățare, precum și aptitudini fundamentale ale învățării colaborative și a rezolvării de probleme. Metodele centrate pe

elev implică individul în evaluarea eficienței procesului de învățare și în stabilirea unor direcții determinate pentru dezvoltarea ulterioară. Aceste avantaje ale învățării centrate pe elev ajută la pregătirea individului atât pentru o integrare mai ușoară în societatea modernă, cât și spre învățarea continuă.

Principiile care stau la baza învățării eficiente centrate pe elev sînt:

- Activitatea de învățare este realizată preponderent de persoana instruită, nu de profesor.
- Acceptarea instruirii tradiționale în calitate de *unul dintre instrumentele* care pot fi utilizate pentru a-i ajuta pe elevi să învețe.
- Acceptarea rolului profesorului ca administrator al procesului de învățare al elevilor.
- Acceptarea faptului că, în mare parte, procesul de învățare nu are loc în sala de clasă și nici cînd cadrul didactic este de față.
- Profesorii trebuie să încurajeze și să faciliteze implicarea activă a elevilor în planificarea și administrarea propriului lor proces de învățare prin proiectarea structurată a oportunităților de învățare atât în sala de clasă, cât și în afara ei.
- Elevii pot învăța individual, în mod eficient în moduri foarte diferite, în condiții maximal adaptate la cerințele și necesitățile individuale.

Exemple de activități de învățare centrată pe elev:

- Lecția pleacă de la experiențele elevilor și cuprinde întrebări sau activități care să îi implice pe elevi.
- Elevii sînt lăsați să aleagă singuri modul cum se informează pe o anumită temă și cum prezintă rezultatele studiului lor.
- Elevii pot beneficia de meditații, în cadrul cărora pot discuta despre preocupările lor individuale cu privire la învățare și pot cere îndrumări.
- Aptitudinea elevilor de a găsi singuri informațiile căutate este dezvoltată – nu li se oferă informații standardizate.
- Pe lîngă învățarea specifică disciplinei respective, li se oferă elevilor ocazia de a dobîndi aptitudini fundamentale transferabile, cum ar fi aceea de a lucra în echipă.
- Se fac evaluări care permit elevilor să aplice teoria în anumite situații din viața reală, cum ar fi studiile de caz și simulările.
- Lecțiile cuprind o combinație de activități, astfel încît să fie abordate stilurile pe care elevii le preferă în învățare (vizual, auditiv, practic/kinetic).
- Lecțiile înlesnesc descoperirile făcute sub îndrumare și solicită participarea activă a elevilor la învățare.

- Lecțiile se încheie cu solicitarea adresată elevilor de a reflecta pe marginea celor învățate, a modului cum au învățat, și de a evalua succesul pe care l-au avut metodele de învățare în cazul lor.

Avantaje:

1. Creșterea motivației elevilor , deoarece aceștia sînt conștienți că pot influența procesul de învățare;
2. Eficacitatea sporită a învățării și a aplicării celor învățate, deoarece aceste abordări folosesc învățarea activă;
3. Învățarea capătă sens deoarece *a stăpîni* materia înseamnă a o înțelege;
4. Posibilitate mai mare de includere – poate fi adaptată în funcție de potențialul fiecărui elev, de capacitățile diferite de învățare, de contextele de învățare specifice.

În particular, la Informatică, centrarea pe elev este realizată prin intermediul activităților de învățare corect selectate, folosirea metodelor interactive de învățare (rezolvare de problem, proiecte de cercetare, proiecte interdisciplinare, studii de caz etc.), activitățile în afara orelor de clasă (instruirea mixtă, la distanță, independentă), individualizarea învățării și a instrumentelor de învățare (lucrul individual la calculator, diversificarea resurselor etc.)

5.6. Utilizarea TIC

În contextul implementării unui sistem modern de instruire, centrat pe elev și axat pe formarea la elevi a competențelor și a capacității de învățare continuă, Tehnologiile Informaționale și Comunicaționale (TIC) devin instrumentul principal pentru selectarea suportului de învățare, exersare, evaluare (autoevaluare) și comunicare în scopul progresului educațional.

Procesul educațional nu mai este limitat de orele de aflare la lecții sau de localul școlii. Elevul este treptat introdus într-un *mediu de învățare*, care este unul adaptiv la necesitățile și particularitățile psihice, de vîrstă, de percepere ale elevului. Este de așteptat că principala componentă a mediului de învățare va fi cea digitală, iar formarea mediului va începe de la activitățile interactive asociate disciplinei Informatica.

Prin *Mediu de învățare* se înțelege spațiul în care cei instruiți realizează activități de învățare și sprijin reciproc, folosind o varietate de instrumente și resurse de informare în contextul eforturilor de realizare a finalităților de învățare. Extinderea mediului de învățare prin utilizarea tehnologiilor moderne generează un model care capătă o serie de caracteristici absolut noi și permite implementarea eficientă a metodelor de instruire, centrate pe elev, cum ar fi, de exemplu, instruirea în bază de problemă, instruirea în bază de proiect, studiile de caz, instruirea colaborativă etc. Astfel, mediul virtual de învățare este definit ca o *aplicație sau*

sistem software care permite rularea integrală sau parțială a unei discipline într-un mediu digital.

Mai multe platforme web 2.0 conțin instrumente software pentru crearea și dezvoltarea mediilor de învățare.

Instrumente Google pentru mediul individual de învățare:

Google mail – caracteristicile contului de mail Google sînt: *spațiul extins acordat* (circa 7 GBt)², posibilitatea de *căutare în mail*, *etichetarea mesajelor*, *arhivarea mesajelor*, formarea *grupurilor de mail* și a *conversațiilor*, chat și forum încorporat. Instrumentul asigură realizarea tuturor cerințelor comunicative înaintate unui mediu de învățare. Atît la disciplina Informatica, cît și la alte discipline școlare poate fi folosit ca un instrument de comunicare sincronă și asincronă.

Google docs – componenta Google care permite *crearea, redactarea, formatarea* în comun, la distanță a documentelor text, *tiparul*, precum și *exportul* lor în cele mai populare formate pentru documente. Aceleași operații pot fi realizate și asupra *foilor de calcul*, *prezentărilor electronice*, *formularelor* (docs.google.com). În contextul utilizării metodelor moderne, interactive de instruire poate fi utilizat pentru elaborarea în comun a proiectelor școlare, studiilor de caz.

Google sites – componenta Google care permite crearea paginilor Web și a aplicațiilor încorporate (ex. Google Calendar, Directions). Paginile Google permit amplasarea în ele a obiectelor de tip *text*, *image*, *video*, *sunet*, *calendar* personal, *hărți*, organizarea unor sarcini personale sau colective pentru realizare (sites.google.com). Formează un mediu ideal pentru crearea și dezvoltarea portofoliilor digitale de învățare, stocarea resurselor educaționale, referințelor la resurse externe.

Google blogs – componenta Google care permite căutarea în blogurile înregistrate Google, crearea blogurilor proprii (blogsearch.google.com). În consecință, se obține încă un instrument performant de comunicare, care permite plasarea resurselor, organizarea discuțiilor, comentarii etc.

Cu doar patru dintre aplicațiile Google poate fi creat un mediu de învățare prietenos, multifuncțional și *extensibil*. Posibilitățile de extindere a mediului de învățare în baza platformei Google se datorează unui set de instrumente speciale, create și dezvoltate în scopuri educaționale.

Google scholar – componentă de căutare în locațiile verificate – biblioteci științifice online. Căutarea poate fi realizată după diverse criterii (titlu, autor, cuvinte cheie etc.). Aplicația poate fi folosită și pentru identificarea tentativelor de plagiat (scholar.google.com).

² În perioada elaborării prezentului ghid.

Google books – biblioteca Google. Conține ediții digitale și cu drept de acces parțial sau integral. De asemenea se conțin informații despre posibilitatea de cumpărare a edițiilor, datele complete despre autori și editori, alte informații relevante. (books.google.com)

Google translate – translatorul Google. Permite traducerea în cele mai diverse limbi a cuvintelor și textelor. Oferă posibilitatea de traducere automată a paginilor Web. Pentru cuvintele solitare propune liste de sinonime. Popularitatea aplicației crește exponențial, transformând *Google translate* în cel mai popular translator online din lume. (translate.google.com)

Google toolbar – componentă de instrumente universale, organizate în formă de bară a programelor de navigare Web. Conține o selecție de instrumente pentru căutare, reținere referințe, comentarii și mesaje Web, wiki, verificare ortografică și traducere etc. (toolbar.google.com)

Google maps – componentă de instrumente universale geografice, hărți. Asigură rezolvarea problemelor de căutare și localizare geografică, selectare a traseului optim, punctelor de interes etc. (maps.google.com)

Toate resursele identificate, create sau redactate cu ajutorul instrumentelor Google, pot fi distribuite către alți membri ai grupului. Ceea ce face ca Google să fie un instrument colaborativ universal este disponibilitatea aplicațiilor în limbi naționale, printre care și limba română, precum și un sistem de asistență și instruire online. Pentru a căpăta acces la instrumentele colaborative Google este suficient un cont de mail Google. Utilizarea instrumentelor Google este simplă și nu necesită cunoștințe prealabile speciale.

Beneficiile utilizării mediului de învățare pe platformă Google:

- Platforma Google reprezintă un sistem de instrumente integrate. Legăturile între componentele de planificare a activităților de învățare, comunicare pentru cunoaștere, evaluare sau feedback, stocare și organizare internă a resurselor proprii și a referințelor către resurse externe se realizează automat, asigurând un confort psihologic în cadrul mediului de învățare.
- Accesul la toate instrumentele platformei Google este realizat printr-un cont unic de poștă electronică. Instalarea componentelor suplimentare nu necesită cunoștințe speciale în domeniul tehnologiilor informaționale și comunicaționale. Interfața aplicațiilor este una sugestivă, simplă și comodă în utilizare.
- Platforma Google asigură posibilitatea utilizării colaborative, online a aplicațiilor de oficiu: editor de text, aplicație de calcul tabelar, pre-

zentări electronice, editor grafic. Astfel procurarea și instalarea pe calculatorul local a versiunilor comerciale ale unor asemenea aplicații devine opțională.

- Resursele educaționale externe în cadrul platformei (biblioteca științifică Google, cărțile Google, hărțile Google) reprezintă date veridice, care pot fi utilizate în calitate de suport științific și didactic.
- Aplicațiile Google sînt în distribuție liberă, gratuită. Utilizarea lor individuală în scopuri educaționale este perfect legală.
- Platforma asigură trei nivele distincte de acces la datele personale: acces personal (doar pentru deținătorul datelor), acces de grup (pentru membrii unui grup sau a unei comunități de învățare) și acces public – pentru toți utilizatorii din rețea, interesați de tematica resurselor.

6. Strategii de evaluare

6.1. Evaluarea axată pe competențe

Dezvoltarea și modernizarea curriculumului școlar duce în mod inevitabil și la modificarea procesului de evaluare. Dacă în curriculumul axat pe conținuturi funcția dominantă a evaluării era de a măsura rezultatele școlare ale elevilor, în curriculumul axat pe atingerea unor obiective evaluarea devine preponderent măsura congruenței rezultatelor școlare cu obiectivele educaționale, atunci la orientarea procesului educațional către obținerea de către elevi a competențelor (în baza curriculumului modernizat) evaluarea devine *o apreciere, ca emitere de judecăți de valoare despre ceea ce a învățat și cum a învățat elevul, pe baza unor criterii precise, bine stabilite anterior* [15, p. 4]. Astfel, în funcție de orientarea curriculară în evaluare este evidențiată una dintre etapele sale, celelalte având un rol complementar.

Pornind de la principiul „comparativ” (de a compara și de a clasifica elevii), evaluarea trece la principiul „criterial” (furnizează informații funcționale, raportând elevii la atingerea obiectivelor comune și oferă soluții de ameliorare), după care acceptă principiul „corectiv” (care oferă elevului informații suplimentare în funcție de dificultățile constatate pentru a-i facilita învățarea).

Axarea procesului educațional pe formarea competențelor școlare a impus elaborarea unui nou principiu general de evaluare: evaluarea „conștientizată” sau „formatoare”, bazată pe psihologia cognitivă și integrarea evaluării în procesul de învățare. Ea oscilează între *învățare*, ca ansamblu al proceselor prin intermediul cărora elevul achiziționează și utilizează cunoașterea, și *învățare de a învăța*, ca proces de „cunoaștere despre autocunoaștere”. Evaluarea formativă corespunde unui demers dominant pedagogic care favorizează participarea activă și autonomia elevului, furnizându-i repere explicite, pentru a dirija propria transformare, ținând cont de propriile dificultăți și lacune.

Din perspectiva abordării curriculare și sistemice a procesului de învățământ, evaluarea face parte integrantă dintr-un tot întreg, ea nu trebuie cercetată separat, dar în corelație cu celelalte activități prin care se realizează procesul de învățământ, cu predarea și învățarea. Pe de altă parte, metodele, tehnicile și instrumentele de evaluare nu pot fi disociate de celelalte variabile ale evaluării: obiectul evaluării (ce se evaluează), criteriile/obiectivele educaționale, strategiile evaluative etc.

În concluzie, evaluarea modernă are un caracter formativ, de apreciere în baza unor criterii bine determinate (standarde), se încadrează organic în activitățile de învățare, accentuând centrarea pe elev a întregului proces educațional.

6.2. Tipuri de evaluare

Există mai multe criterii de stabilire a tipologiei evaluării. Cel mai des utilizat este criteriul scopului și frecvenței, în funcție de care distingem:

Evaluarea inițială, prin care se stabilește nivelul de pregătire al elevului la începutul unei perioade sau etape de lucru, la începutul unei unități de învățare, teme, capitol, precum și condițiile în care acesta se poate integra în programul de instruire. Se recomandă utilizarea testării la calculator, cu ajutorul testelor locale sau online, fără aplicarea notării elevilor.

Evaluarea continuă (formativă), presupune verificarea permanentă a rezultatelor, pe tot parcursul procesului de instruire, de obicei operându-se pe secvențe mici. Trecerea la secvența următoare se realizează numai după ce se cunoaște modul de desfășurare și eficiența educațională a secvenței evaluate, rezultatele obținute de elevi, prin adoptarea de măsuri de ameliorare privind procesul de învățare și performanțele unor elevi. Poate fi realizată individual sau în grup, în baza unor interviuri orale, rezolvării de exerciții și probleme la calculator, dar și a realizării unor etape ale proiectelor.

Evaluarea sumativă (cumulativă) se realizează la sfârșitul unei etape de instruire, la finele studierii unei unități de învățare, teme, capitol și, periodic, la sfârșitul semestrelor, al anului școlar, al ciclului de școlarizare, prin conceperea unor subiecte cuprinzătoare care să acopere întreaga arie tematică abordată. La finalul unității de învățare poate fi aplicată testarea online sau locală la calculator. Pentru organizarea tezelor semestriale se vor folosi metodologiile indicate în documentele normative. Organizarea testării naționale (examenul de bacalaureat la Informatică) este realizată în concordanță cu programul pentru examen la disciplină, iar forma de organizare este stabilită de documentele normative. Prin urmare, pregătirea elevilor pentru examenul de bacalaureat va fi efectuată în concordanță cu programul de examen la disciplina Informatica.

Dinamica rezultatelor evaluării pe parcursul unui interval de timp (perioade de învățare) permite și determinarea progresului elevului în formarea competențelor specifice, dezvoltate de disciplină în acea perioadă.

6.3. Metode și tehnici de evaluare

Metode de evaluare

Metoda de evaluare este o cale prin care profesorul oferă elevilor posibilitatea de a demonstra nivelul de stăpânire a cunoștințelor, de formare a abilităților, competențelor specifice testate prin utilizarea unei diversități de instrumente adecvate obiectivului de evaluare propus.

Metodele tradiționale de evaluare (utilizate în pedagogie o perioadă îndelungată de timp). Dintre ele pot fi menționate următoarele: evaluarea orală, evaluări scrise, probe practice (de obicei realizate la calculator).

Verificarea orală este realizată în baza unei conversații prin care profesorul urmărește identificarea achiziției de cunoștințe. Conversația poate fi individuală, frontală sau combinată. Avantajul principal este feed-back-ul imediat. Metoda favorizează și dezvoltarea capacităților de exprimare ale celor instruiți. Neajunsul metodei este dependența de o multitudine de factori care pot influența răspunsul elevului: starea de moment a profesorului/elevului, gradul diferit de dificultate a întrebărilor, starea psihică sau particularitățile de caracter ale celor evaluați etc. Se recomandă, în calitate de verificare-expres în cadrul lecțiilor teoretice, nu atât pentru aprecierea elevilor, cât pentru evaluarea calității activităților de învățare din cadrul lecției.

Evaluarea scrisă presupune realizarea unor sarcini pe suporturi scrise, cum ar fi lucrările de control, tezele, examenele naționale. În cadrul acestui tip de evaluare dispăre contactul direct al elevului cu profesorul evaluator, iar în unele cazuri evaluarea este chiar externă, ceea ce permite diminuarea elementului subiectiv în evaluare. Printre avantajele metodei se numără posibilitatea verificării unui număr relativ mare de elevi într-un interval de timp redus, verificarea și notarea în conformitate cu criterii strict determinate apriori (barem, schema de notare), crearea unor condiții optime pentru elevii cu deficiențe de exprimare orală etc. În procesul educațional intern al instituției de învățământ este recomandată evaluarea scrisă pentru teze semestriale și lucrări de control la finalizarea unităților de învățare (dacă nu există condiții pentru realizarea testărilor asistate de calculator).

Proble practice de evaluare presupun verificarea funcționalității cunoștințelor achiziționate, precum și a interiorizării/exteriorizării lor. În cadrul disciplinei Informatica acest tip de evaluare se realizează prin rezolvare de probleme (prin elaborarea programelor proprii sau utilizarea instrumentelor online) la calculator. Se încadrează în activitățile de învățare prin rezolvare de probleme, la etapa finală – explicarea rezolvării și validarea rezultatelor.

La utilizarea metodelor tradiționale de evaluare trebuie menținut un echilibru între probele orale, scrise și cele practice.

Metode complementare (moderne de evaluare), formative, orientate spre individualizarea educației prin centrarea pe elev și dezvoltarea acestuia sînt de asemenea integrate în activitățile de învățare pe termen mediu (1 – 2 săptămîni) sau lung (pînă la un semestru) și reprezintă aprecierea finală sau intermediară a succesului realizării activităților de către elev/grup. Printre cele mai recomandate metode complementare se numără *referatul, studiul de caz, proiectele tematice și de cercetare, portofoliul elevului la disciplină, autoevaluarea*.

Criteriile de evaluare pentru metodele complementare vor fi elaborate în baza indicatorilor de performanță stabiliți în corespundere cu standardele de competență la disciplina Informatică, taxonomiile adecvate (Bloom, Simpson, Dove, Krathwohl), gradul de complexitate și durata activităților evaluate. Astfel, referatul și studiul de caz pot fi evaluate la finalizarea acestora, pe când proiectele, portofoliile elevilor pot fi evaluate pe măsura realizării lor, la finalizarea unor etape specificate apriori.

Tehnici de evaluare

Tehnica de evaluare reprezintă modalitatea prin care cadrul didactic declanșează și orientează obținerea din partea elevilor a răspunsurilor la itemii formulați. O probă cu mai mulți itemi poate solicita una sau mai multe tehnici de evaluare, în funcție de tipurile de itemi pe care îi include.

Pentru testările realizate prin metode tradiționale (orală, scrisă, la calculator) se vor utiliza diferite tipuri de itemi, preponderent:

Itemi obiectivi: itemi cu alegere multiplă, itemi cu alegere duală, itemi tip pereche:

Itemii cu alegere multiplă presupun un enunț și o multitudine de variante de răspuns, dintre care doar una este corectă și urmează să fie selectată în modul indicat în enunț. Se folosesc pentru măsurarea rezultatelor învățării de nivel taxonomic inferior: măsurarea cunoștințelor acumulate de elevi, cunoașterea terminologiei, elementelor/faptelor științifice, principiilor metodelor și procedurilor. La nivel mediu și superior (funcționalizare, interiorizare) pot fi utilizate pentru evaluarea abilităților de a identifica aplicații ale faptelor și principiilor, a interpreta relația cauză-efect, a argumenta metode și proceduri.

Exemplu: Numărul zecimal 124 în sistemul de numerație cu baza 8 se va scrie ca:

- a) 174 b) 172 c) 164 d) F6

Itemii cu alegere duală solicită elevului să selecteze din două răspunsuri posibile: adevărat/fals; corect/greșit; da/nu; acord/dezacord etc. Ei pot fi utilizați pentru recunoașterea unor termeni, date concrete, principii, diferențiere între enunțuri bazate pe fapte sau de opinie, la nivel taxonomic inferior.

Exemplu: Stabiliți valoarea de adevăr a afirmației: Sistemul zecimal de numerație (în baza cifrelor arabe) este unul NEPOZIȚIONAL

- a) ADEVĂRAT b) FALS

Itemii perechi solicită stabilirea unor corespondențe/asociații între cuvinte, propoziții, fraze, litere sau alte categorii de simboluri dispuse pe două coloane. Elementele din prima coloană se numesc *premise*, iar cele din a doua coloană reprezintă *răspunsurile*. Criteriul sau criteriile pe baza cărora se stabilește răspun-

sul corect sînt enunțate/explicitate în instrucțiunile care preced cele două coloane. Se limitează, de obicei, la măsurarea informațiilor de facto, bazîndu-se pe simple asociații, pe abilitatea de a identifica relația existentă între două lucruri/noțiuni/simboluri etc. Itemii tip pereche pot propune diverse tipuri de relații: termeni/definiții; reguli/exemple; simboluri/concepte; principii/clasificări etc. Se poate utiliza material pictural sau o reprezentare grafică.

Exemplu: Stabiliți prin săgeți corespondența dintre noțiunile din coloana stîngă și definițiile din coloana dreaptă:

<i>Noțiuni</i>	<i>Definiții</i>
Eroare absolută	raportul dintre eroarea absolută Δ și modulul numărului exact A ($A \neq 0$)
Eroare relativă	modulul diferenței dintre valoarea exactă A și valoarea aproximativă a

Itemi semiobiectivi: itemii cu răspuns scurt, itemi de completare, întrebări structurate.

Itemii cu răspuns scurt se divizează în cîteva subcategorii distincte:

- Întrebearea clasică, care vizează un răspuns așteptat (unic sau multiplu);
- Exercițiul, rezolvarea căruia generează un rezultat;
- Textul indus, mai puțin aplicabil la disciplina Informatică, datorită structurii sale.

Exemplu: Item cu răspuns scurt, multiplu:

Care este tehnica de programare, ce poate fi folosită pentru rezolvarea exactă a problemei rucsacului? (Răspuns posibil: Triere, Reluare)

Itemi de completare: enunț în care se omit unul sau mai multe cuvinte importante. Se cere elevului să găsească acele cuvinte și să le scrie.

Exemplu: Item de completare:

La transformarea directă a unui număr binar întreg în sistemul de numerație cu baza 8, acesta se divide în grupe a cîte _____ cifre pornind de la _____.

Întrebările structurate sînt formate din mai multe subîntrebări de tip obiectiv, semiobiectiv sau minieseu legate între ele printr-un element comun. Ele ocupă nișa dintre tehnicile de evaluare cu răspuns liber (deschis) impuse de itemii subiectivi și cele cu răspuns limitat (închis) impuse de itemii de tip obiectiv. Întrebările structurate pot fi folosite pentru evaluarea cunoașterii principiilor tehnicilor de programare sau a metodelor de calcul numeric, structurilor de date etc. Subîn-

trebările pot viza practic toate categoriile taxonomice, pornind de la simpla reproducere (definiții, enumerări etc.) până la aplicarea cunoștințelor, analiză, sinteză și formularea de ipoteze, judecăți de valoare.

Exemplu: Întrebare structurată:

Descrieți condițiile pentru existența soluției ecuației $f(x)=0$ pe segmentul $[a, b]$

Subîntrebări: De ce este necesar ca funcția să fie continuă? De ce este necesar ca semnul funcției în extremități să fie diferit? Ce se poate spune despre numărul de soluții pe segment?

Itemi subiectivi sau cu răspuns construit elaborat/dezvoltat. Sînt itemi largi, care solicită răspunsuri a căror realizare mobilizează cunoștințe și abilități ce iau forma unor structuri integrate. Principalele forme de itemi subiectivi sînt: itemii cu răspuns construit scurt, puțin elaborat, itemii tip rezolvare de probleme, itemii tip eseu, itemii cu răspuns dezvoltat.

Itemii cu răspuns construit scurt, puțin elaborat cer elevului să exprime o idee importantă în cîteva fraze: o explicație, o relație, o definiție descriptivă, algoritm elementar, model matematic etc. Profesorul fixează criterii exacte cu privire la conținutul răspunsului, extensie, structură etc.

Itemii tip rezolvare de probleme. Rezolvarea de probleme reprezintă antrenarea elevului într-o activitate nouă, diferită de activitățile curente propuse anterior. Scopul principal: dezvoltarea creativității, a gândirii divergente, a gândirii algoritmice, a imaginației, a capacității de a generaliza, de a reformula o problemă etc. În funcție de problemă se verifică componentele competențelor, formate la etapele de funcționalizare, interiorizare și exteriorizare. Rezolvarea problemei la calculator presupune realizarea unor etape distincte, care pot fi evaluate separat, în scopul determinării mai exacte a progresului elevului.

Itemii tip eseu: permit evaluarea globală din perspectiva unei competențe care nu poate fi evaluată eficient cu ajutorul itemilor obiectivi sau semiobiectivi. Se pune în valoare abilitatea de organizare și integrare a ideilor, exprimarea personală în scris, abilitatea de a interpreta și aplica datele. Itemul tip eseu cere elevului să construiască, să producă un răspuns liber în conformitate cu un set de cerințe date. La orele de informatică poate fi propusă realizarea unui mini eseu structurat sau semistrukturat cu răspuns așteptat ordonat și orientat la teme preponderant descriptive: structura calculatorului, topologia rețelelor de calculatoare, servicii Internet etc.

Itemii cu răspuns dezvoltat presupun un volum extins al răspunsului și nu sînt de obicei aplicabili la disciplina Informatică.

6.4. Forme de evaluare curentă

Evaluarea curentă (formativă sau de progres) se aplică pe tot parcursul desfășurării procesului educațional. Cu ajutorul ei se menține un flux evaluativ continuu, care urmează să asigure un progres permanent. Prin specificul ei, evaluarea formativă este o evaluare centrată pe activități educaționale, destinată rectificării, reajustării și adaptării acestora, ameliorării și optimizării, reglării și autoreglării învățării și predării. În felul acesta îndeplinește o funcție formativă de rafinare și dezvoltare a potențialului de învățare, dar și a celui de predare, de formare sau dezvoltare propriu-zis.

Urmind principiul feed-back-ului, tehnicile de evaluare formativă contribuie la organizarea condițiilor de succes al învățării, înscriindu-se în logica unei pedagogii centrate pe succesul unui cât mai mare număr de elevi.

Forme:

Observare curentă a comportamentului școlar al elevului: poate fi realizată practic în orice moment al activității educaționale pentru a evalua performanțele elevilor, dar mai ales pentru a evalua comportamente afectiv-atitudinale.

Caracteristicile care pot fi evaluate sînt:

La nivel de concepte și capacități:

- organizarea și interpretarea datelor;
- selectarea și organizarea corespunzătoare a instrumentelor de lucru;
- descrierea și generalizarea unor procedee, tehnici, relații;
- utilizarea materialelor auxiliare pentru a demonstra ceva;
- identificarea relațiilor;
- utilizarea calculatorului în situații corespunzătoare.

La nivel atitudinal:

- concentrarea asupra sarcinii de rezolvat;
- implicarea activă în rezolvarea sarcinii;
- punerea unor întrebări pertinente profesorului;
- completarea, îndeplinirea sarcinii;
- revizuirea metodelor utilizate și a rezultatelor.

Interogarea orală. O formă de evaluare individuală expres, care face parte din cele tradiționale și este puțin eficientă într-un sistem educațional centrat pe elev, unde prioritate o are discuția didactică, în care se implică un număr mai mare de elevi. Pentru evaluarea elevilor în cadrul discuției poate fi folosită *observarea curentă*.

Rezolvare de exerciții și probleme. Este o activitate educațională practică de funcționalizare/interiorizare a cunoștințelor obținute recent, care finalizează cu

un rezultat ce poate fi evaluat (corect/incorect). Suplimentar are loc evaluarea procesului de rezolvare și identificarea „punctelor slabe” la fiecare etapă de rezolvare. Atât exercițiile, cât și problemele pot fi tradiționale (pentru rezolvare scrisă) și inovaționale (pentru rezolvare la calculator).

Teste grilă. Teste scrise sau asistate de calculator (local sau online). Se propun elevilor (selectiv sau frontal) la etapa finală a lecției sau activității educaționale. Sînt elaborate în bază de itemi obiectivi și semiobiectivi și se referă la cunoștințele achiziționate pe parcursul activității. Este preferabilă organizarea testelor grilă în baza unor aplicații specializate (AeL, LeMill, eFront sau chiar în aplicații de calcul tabelar), ceea ce permite obținerea unui feed-back imediat. Totuși, utilizarea aplicațiilor specializate este preferabilă din motivul unei diversificări mai extinse a tipologiei itemilor (AeL, LeMill,) a realizării automate a statisticilor și posibilității individualizării testelor prin formarea unei bănci de itemi.

Exemplu: fragment de test, elaborat pentru evaluarea curentă la tema sisteme de numerație, cl. X pe platforma online LeMill:

Activități R. A. I. (Răspunde – Aruncă – Interoghează) stimulează capacitățile elevilor de a comunica (prin întrebări și răspunsuri) cunoștințele achiziționate recent. Modelul de organizare este următorul: la sfîrșitul unei lecții sau al unui moment de lecție, profesorul, împreună cu elevii, investighează rezultatele predării-învățării, printr-un joc de aruncare a unei mingii mici de la un elev la altul. Cel care aruncă mingea trebuie să formuleze o întrebare din materia învățată recent pentru cel care o prinde. Cel care prinde mingea răspunde la întrebare, apoi preia funcția aruncătorului. Condiții de eliminare: necunoașterea răspunsului la întrebarea primită sau formulată. Eliminarea

The screenshot shows the LeMill online platform interface. The top navigation bar includes 'Content', 'Methods', 'Tools', and 'Community'. The main content area displays a math problem titled 'sisteme de numerație'. The problem text is: 'Numărul de cifre distincte în sistemul de numerație cu baza 8 este'. Below the text are five radio button options: 8, 3, 16, 7, and 10. The second question is: 'Sistemul decimal de numerație (în baza cifrelor arabe) este un NEPOZITIONAL'. Below this text are two radio button options: DECIMAL and FALSE. The third question is: 'Numărul decimal 124 în sistemul de numerație cu baza 8 se va scrie ca:'. Below this text are five radio button options: 16, 174, 172, 164, and 124. The fourth question is: 'La transformarea directă a unui număr binar într-un sistem de numerație cu baza 8, acesta se divide în grupe a câte [] cifre pornind de la []'. Below this text are two input fields. At the bottom, there is a checkbox labeled 'If you want the system to check your answers and send feedback as e-mail to you or your teacher, you need to fill this e-mail form.' and a 'System' button. The footer contains links: 'About | Help | FAQ | LeMill development site | Feedback | Report a problem'.

continuă conduce treptat la rămînerea în grup a celor mai bine pregătiți. Metoda poate fi organizată cu toată clasa sau pe grupe mici, separate, în ultimul caz fiind organizată o etapă de „final” cu învingătorii din fiecare grup. Compartimentele la care poate fi aplicată metoda sînt cele preponderent descriptive: structura calculatorului, rețele de calculatoare, elemente de web design etc.

Activități 3-2-1 (*3 termeni (concepte) – 2 idei – o capacitate, pricepere sau abilitate*). Metoda este folosită pentru a aprecia rezultatele unei secvențe didactice sau ale unei activități. Elevii (selectați sau întreaga clasă) vor scrie: *3 termeni (concepte)* din ceea ce au învățat, *2 idei* despre care ar dori să învețe în continuare și *o capacitate, pricepere sau abilitate* pe care consideră că au obținut-o pe parcursul activităților de predare-învățare.

Avantajul principal este faptul că elevii devin conștienți și responsabili de rezultatele obținute, apar elemente de implicare a lor în procesul educațional (ce ar vrea să știe în continuare) și de autoevaluare (ce abilitate au obținut).

O descriere mai amplă a formelor de evaluare poate fi găsită în edițiile specializate, atât tradiționale [15], cît și digitale³ [www.edu.ro/index.php/articles/3909]

6.5. Evaluarea centrată pe succes

Una dintre funcțiile prioritare ale evaluării este cea de suport al elevului pentru obținerea unui progres maxim al învățării. Evidențierea acestei funcții transformă evaluarea din formativă în evaluare formatoare, orientată pentru asigurarea succesului maximal al învățării⁴.

Evaluarea formativă cercetează procesul de învățare, realizat în sala de clasă și este centrată pe intervenția profesorului către elev/clasă. Ea implică răspunsuri din partea elevului, la intervenția profesorului. *Evaluarea formatoare* vine către elev și inițiativa îi aparține acestuia din urmă. Evaluarea formativă facilitează învățarea, profesorul conduce elevul către atingerea unor obiective, scopuri. În evaluarea formatoare inițiativa de învățare și implicit de evaluare aparține elevului, prin reflecții asupra rezultatelor activității sale. Rolul profesorului se schimbă: el urmărește evoluția elevului, acordînd consultanță în momentele necesare.

Evaluarea formativă nu garantează de la sine succesul elevului, deoarece vine din exterior. Rolul său prioritar este cel de organizare eficientă a procesului de predare-învățare cu ajutorul profesorului. În același timp evaluarea formatoare are un grad mai mare de eficiență, deoarece inițiativa evaluării vine de la cel in-

³ Sursă activă la momentul scrierii prezentului ghid.

⁴ Nunziati, G. (1990), „Pour construire un dispositif d'évaluation formative”, *Cahiers Pédagogiques*, No. 280, pp. 48-64.

struit, care cere o apreciere externă a succesului său, ceea ce presupune și o etapă prealabilă de autoevaluare și coevaluare.

Pentru realizarea evaluării orientate către succes, profesorul trebuie să utilizeze formele și metodele moderne de instruire, să dețină tehnici prin care să creeze la elevi competența de învățare independentă, astfel încât ei să ia o atitudine activă față de procesul de învățare și să-și poată evalua propriile activități. Prin urmare, evaluarea formatoare gestionează și autogestionează activitățile de învățare atât prin intermediul profesorului, cât și prin intermediul elevului.

Caracteristicile dominante ale evaluării centrate pe succes:

- inițiativa evaluării aparține elevului și este orientată sau nu de către profesor;
- rezultă din analiza de către elev a propriei transformări, învățări, a propriului progres;
- elevul dispune de o motivație interioară pentru evaluare;
- are un impact pozitiv asupra relației elev-profesor.

Fiind aplicată în cadrul disciplinei Informatica, evaluarea formatoare se realizează prin includerea în instrumentarul de evaluare a profesorului a metodelor complementare de învățare-evaluare, centrate pe elev: instruirea în bază de probleme, referate, studii de caz, proiecte tematice și de cercetare, portofoliul etc. Se recomandă trecerea treptată de la activitățile mai simple, familiare elevilor (referat, rezolvare de probleme) către activitățile extinse în timp, complexe din punct de vedere al instrumentarului folosit și al finalităților propuse (studiul de caz, proiectul, portofoliul).

6.6. Materiale didactice suport pentru evaluarea competențelor

Materialele propuse în acest compartiment vizează două aspecte distincte ale evaluării, importante în special pentru disciplina Informatică:

- Suport teoretic și metodologic pentru evaluările de competențe prin metode tradiționale;
- Suport teoretic și metodologic pentru evaluările asistate de calculator.

Suport teoretic și metodologic pentru evaluările de competențe prin metode tradiționale:

1. A. Stoica, S. Musteață, *Evaluarea rezultatelor școlare. Ghid metodologic*, Chișinău, Liceum, 1997.
2. P. Lisevici, *Evaluarea în învățământ. Teorie, practică, Instrumente*, Aramis, București, 2002.

3. M. Manolescu, *Evaluarea școlară. Metode, tehnici, instrumente*, Editura Meteor Press, București, 2005.
4. O. Cosovan, A. Ghicov, *Evaluarea continuă la clasă. Ghid metodologic*, Chișinău, Știința, 2007.
5. I. Ciobanu, S. Corlat, A. Malearovici ș.a., *Bacalaureat 2003. Informatică*, Chișinău, Lyceum, 2003.
6. D. Arvunescu, C. Dumitru, C. Gălățan ș.a., *Ghid de pregătire Bacalaureat Informatică intensiv. C/C++/Pascal*, Sigma, București, 2008.
7. C. Bălănescu, E. Cerchez, D. Grigoriu ș.a., *Teste de sinteză în programare. Bacalaureat Informatică*, L&S, București, 2005.
8. Tim S. Roberts, *Self, Peer and Group Assesment in e-Learning*, Idea Group Inc. 2006

Suport teoretic și metodologic pentru evaluările asistate de calculator:

9. Ю. Дорощенко, П. Ротаенко, *Достовірність компютерного тестування*, Педагогічна Думка, Київ, 2007.
10. M. Ilie, R. Jugureanu ș.a., *Manual de instruire a profesorilor pentru utilizarea platformelor de eLearning*, Litera Internațional, 2008.
11. Jason Cole, Helen Foster, *Using Moodle: teaching with the popular open source course management system*, O'Reiley Media Inc., 2007.

7. Recomandări metodice de utilizare a echipamentului și materialelor didactice, a manualelor existente și noi în procesul de implementare a curriculumului modernizat

Procesul didactic, axat pe formarea competențelor, centrat pe elev presupune o modificare considerabilă a tehnicilor de selectare a materialelor didactice, manualelor și resurselor educaționale, utilizate de profesori. Pe de o parte se extinde gradul de libertate a profesorului în selectarea acestor resurse, pe de alta – accesul la resursele în distribuție liberă (pentru toate disciplinele de studii și în special pentru disciplina Informatica) se extinde în permanență, iar numărul și calitatea resurselor se află în continuă creștere.

O problemă de primă importanță în această situație este selectarea resurselor de calitate, în special a celor digitale, plasate pe Web. De cele mai multe ori resursele, chiar și calitative, necesită o adaptare (lingvistică, metodologică, curriculară etc.) pînă la momentul de acceptare a utilizării în sala de clasă.

O oportunitate creată de dezvoltarea tehnologiilor Web pentru profesori este posibilitatea de creare a propriilor resurse digitale pentru realizarea procesului de predare-învățare și de îmbunătățire colectivă a resurselor create. Cele mai eficiente rezultate pot fi obținute în elaborarea bateriilor de itemi pentru evaluări online, care pot fi discutate, testate în prealabil și corectate de un grup de profesori, și abia mai apoi propuse pentru utilizare elevilor.

Pentru a asigura procesul instructiv la disciplină se recomandă următoarele manuale, ghiduri, ediții specializate, resurse digitale verificate și medii pentru crearea propriilor resurse:

A. Manuale și ghiduri aprobate de Ministerul Educației (clasele X – XII)

Manuale

1. Anatol Gremalschi, Iurie Mocanu, Ludmila Gremalschi, *Informatică. Manual pentru clasa a 10-a*, Știința, Chișinău, 2000 sau 2007.
2. Anatol Gremalschi, Iurie Mocanu, Ion Spinei, *Informatica. Limbajul PASCAL. Manual pentru clasele IX-XI*, Știința, 2000, 2002 sau 2005.
3. Anatol Gremalschi, *Informatica. Manual pentru clasa a XI-a*, Știința, Chișinău, 2008.
4. Anatol Gremalschi, *Informatica. Tehnici de programare. Manual pentru clasa a XI-a*, Știința, Chișinău, 2003.

5. Anatol Gremalschi, Sergiu Corlat, Andrei Braicov, *Informatica. Manual pentru clasa a XII-a*, Știința, Chișinău, 2010.

Ghiduri

1. Vasile Andronic, *Informatica. Ghid de implementare a curriculumului modernizat în învățământul liceal*, Î.E.P. Știința, Chișinău, 2007.
2. *Științe exacte. Matematică. Fizică. Informatică. Curriculum național. Ghid metodologic de implementare pentru învățământul liceal*, C.E. Pro Didactica, Chișinău, 2000.
3. Sergiu Corlat, Lilia Ivanov, *Calculatorul în predare și învățare. Ghid metodologic pentru formarea cadrelor didactice din învățământul pre-universitar*, Î.E.P. Știința, Chișinău, 2007.

B. Ediții recomandate

1. Andrei Braicov, *HTML. Ghid de inițiere*, Chișinău, Prut Internațional, 2008
2. Andrei Braicov, *Turbo Pascal. Culegere de probleme*, Chișinău, Prut Internațional, 2007
3. Valeriu Cabac, *Elemente de modelare matematică*, Chișinău, Lumina, 1998.
4. Emanuela Cerchez, Marinel Șerban, *Informatica*, Iași, Editura Polirom, 2000.
5. Emanuela Cerchez, Marinel Șerban, *Programarea în limbajul C/C++ pentru liceu. Vol I, II, III*, Iași, Editura Polirom, 2006 – 2008.
6. Sergiu Corlat, Lilia Ivanov, *Calcul numeric. Curs de lecții*, Chișinău, CCRE „Presa”, 2004.
7. Cristian Giumale, *Introducere în analiza algoritmilor*, Iași, Editura Polirom, 2004.
8. Sergiu Corlat, *Algoritmi și probleme de geometrie computațională*, Chișinău, Prut Internațional, 2009.
9. Stiven S. Skiena, Miguel A. Revilla, *Programming Challenges*. Springer Verlag, New York, 2003
10. Thomas Cormen, Charles Leiserson, Ronald Rivest, *Introducere în algoritmi*, Agora, 2001
11. Robert Sedgewick, *Algorithms in C*, Addison Wesley, 2002.
12. Eugenia L. Stamate, *Crearea și publicarea unui site web. Limbajul HTML și elemente de web design*, București, Paralela 45, 2005

Resurse educaționale digitale on-line și locale

Setul de aplicații software poate fi substanțial completat cu resurse care au un caracter educațional: *resurse pasive* (texte, imagini, secvențe video), *resurse interactive*, *aplicații pentru testarea asistată de calculator*; *aplicații de simulare*; *biblioteci*, *dicționare* și *enciclopedii*.

Dezvoltarea tehnologiilor de transmitere a datelor, precum și diversificarea serviciilor de conectare la rețea în Republica Moldova permit conectarea tuturor instituțiilor de învățământ preuniversitar la rețeaua Internet prin bandă largă. În consecință apare posibilitatea de a exploata eficient nu numai resursele educaționale locale, dar și resursele educaționale în distribuție liberă din rețea. Selectarea oricărui tip de resursă/aplicație educațională poate fi realizată simplu cu ajutorul aplicațiilor de căutare, astfel încât profesorul să poată stabili individual criteriile principale de selectare a resurselor. Totuși, vom menționa câteva resurse specializate, cu folosire gratuită, care pot fi utilizate eficient la orele de Informatică:

www.solveymath.com – calculator matematic on-line. Permite trasarea graficelor funcțiilor, calculul integralelor, derivatelor determinanților numerici; rezolvarea ecuațiilor și sistemelor de ecuații etc. Poate fi folosit pentru rezolvarea problemelor de matematică, informatică, fizică în cadrul orelor la discipline sau a proiectelor transdisciplinare.

www.myphysicslab.com – simulator on-line al fenomenelor fizice. Poate fi folosit la lecțiile de fizică, informatică (elemente de modelare), precum și pentru elaborarea proiectelor informatice cu caracter transdisciplinar.

<http://campion.edu.ro> – site pentru pregătirea de performanță a elevilor și profesorilor în domeniul informaticii. Permite participarea la competiții de programare on-line, accesarea resurselor educaționale din domeniul informaticii, lucrul cu arhiva de probleme. Conține opțiunea de evaluare automată a problemelor de programare.

<http://wikipedia.org> – enciclopedie on-line. Conține publicații la diverse subiecte, în particular la științele reale și informatică. Dispune de o bogată colecție de algoritmi fundamentali descriși în diverse moduri, precum și însoțiți de exemple dinamice. Poate fi accesată în diferite limbi, cea mai extinsă fiind versiunea engleză. Pentru traducerea automată a subiectelor care nu se regăsesc în versiunea română poate fi folosit translatorul automat <http://translate.google.com>. Tot el poate fi folosit și pentru traducerea oricărei alte informații din rețea.

<http://books.google.com> – biblioteca de cărți Google. Conține un număr impunător de titluri din cele mai diverse domenii, inclusiv informatică și aplicațiile acesteia, pregătirea de performanță pentru concursurile de programare, pedagogie. Permite căutarea cărților după diferiți parametri, inclusiv în text. Edițiile pot fi vizualizate integral sau parțial, în funcție de dreptul de acces, acordat de sistem.

<http://lemill.net> – portal pentru instruirea mixtă, cu un repozitoriu de resurse incorporat. Permite căutarea și utilizarea resurselor de pe portal, adaptarea acestora, crearea și plasarea resurselor proprii, elaborarea testelor și evaluarea. Pentru resursele în distribuție liberă de pe portal este posibil exportul în standard SCORM pentru integrarea ulterioară în alte sisteme de management al învățării. Conține și un modul de comunicare. Este accesibil în mai multe limbi, inclusiv engleză, rusă.

Repozitorii specializate

Pot fi recomandate următoarele locații web pentru plasarea, păstrarea și schimbul de resurse educaționale:

www.scribd.com – comunitate de autori și specialiști din diverse domenii. Portalul permite accesarea pentru lectură a documentelor scrise cu diferite tematici, în special educație, științe exacte, informatică. Devenind un membru al comunității, căpătați posibilitatea nu numai de a citi resursele, dar și de a le descărca, sau plasa propriile resurse. Conține resurse în diferite limbi, printre care și limba română.

www.slideshare.net – Platformă pentru stocarea prezentărilor electronice. Permite accesarea pentru vizualizare a prezentărilor electronice cu diferite tematici, inclusiv educație, științe exacte, informatică. Devenind un membru al comunității, căpătați posibilitatea nu numai de a vizualiza și descărca prezentările, dar și de a plasa propriile prezentări. Platforma permite căutarea resurselor după cuvinte-cheie, ceea ce ușurează esențial procesul de selectare a resurselor adecvate. Conține opțiunea de a integra prezentările selectate în alte resurse web: site-uri, bloguri. Resursele pentru care este permisă descărcarea sau integrarea se află sub licența CCL.

www.youtube.com – Platformă pentru stocarea și difuzarea secvențelor video. Permite vizualizarea secvențelor video stocate pe platformă. Conține un mare număr de lecții și secvențe video cu conținut educațional informatic: structura calculatorului, structuri de date, subprograme, algoritmi fundamentali, web design, baze de date etc. Devenind un membru al comunității, căpătați posibilitatea nu numai de a vizualiza și secvențele video, dar și de a plasa propriile resurse. În cadrul platformei este organizată căutarea resurselor după cuvinte-cheie, ceea ce ușurează esențial procesul de selectare a resurselor adecvate. Conține opțiunea de a integra secvențele video selectate în alte resurse web: site-uri, bloguri.

Prezența unui număr semnificativ de resurse gratuite on-line și a versiunilor locale ale acestora nu garantează succesul educațional prin utilizarea lor masivă. Resursele digitale sînt doar încă un instrument care este pus la dispoziția profesorului, iar succesul învățării depinde de profesionalismul acestuia, de capacitate lui de a selecta resursele adecvate în funcție de temă, clasă, situație.

8. Deosebiri ale procesului de predare-învățare-evaluare de profiluri

Competența digitală este una dintre competențele generale care sînt vitale în societatea contemporană. În cadrul învățămîntului preuniversitar ea este dezvoltată preponderent prin competențele specifice ale disciplinei Informatică, care se realizează printr-o gamă de conținuturi didactice atît cu caracter fundamental, cît și cu caracter aplicativ. Prin urmare, setul de competențe specifice disciplinei, atît pentru profilul real, cît și pentru profilul umanist, este același.

Diferențele, generate de specificul fiecărui profil, se manifestă în gradul de penetrare a competențelor specifice disciplinei, gradul de penetrare fiind determinat de setul de subcompetențe și conținuturi din care se formează competențele specifice disciplinei.

Suplimentar, specificul activităților didactice cu caracter practic la disciplina Informatică, în special rezolvarea de probleme, presupune dezvoltarea concomitentă a cîtorva subcompetențe și competențe specifice disciplinei.

Exemplu: rezolvarea la calculator a problemei determinării celei mai înalte temperaturi pe parcursul unei perioade. (În expunerea abstractă – a elementului cu valoare maximă într-un tablou.) Competențele specifice, dezvoltate pe parcursul acestei activități sînt: CS2, CS4, CS5, CS6. Dacă în procesul de rezolvare se impun condiții suplimentare asupra stocării datelor inițiale și a rezultatelor, este dezvoltată și competența specifică CS7.

Prin urmare, integrarea reciprocă a competențelor specifice, subcompetențelor și conținuturilor didactice impune o abordare unică a concepției didactice a disciplinei, indiferent de profil. Diferențele de abordare se manifestă doar în conținuturile didactice vizate de curriculum pentru fiecare profil aparte și gradul de penetrare a conținuturilor de învățare în procesul activităților de învățare.

Tabelul 2 conține o sistematizare a activităților de învățare în cadrul procesului de formare a competențelor specifice disciplinei pe etape (fixarea cunoștințelor/funcționalizarea lor/interiorizarea/exteriorizarea), cu diferențiere după profiluri și anii de studii.

Tabelul 2. Particularizări ale conținuturilor și activităților pe etapele de formare a competențelor după profil

Clasa X

Detalii de conținuturi și activități pe etapele de formare a competențelor după profil

Conținuturi	Fixare cunoștințe	R	U	Funcționalizare	R	U	Interiorizare	R	U	Exteriorizare	R	U	Observații
Tipuri de date structurate													
Declarații de tipuri de date structurate; Formule meta-lingvistice și diagramele sintactice ale declarațiilor de tipuri de date structurate.	Exerciții de definire și clasificare a tipurilor de date structurate (tablou, șir de caractere, articol, mulțime, fișier), folosind formulele meta-lingvistice și diagramele sintactice ale tipurilor de date în studiu.	+	+	Exerciții de declarare a variabilelor de tip tablou, șir de caractere, articol, mulțime, fișier.	+	+	Identificarea situațiilor din viața reală în care apare necesitatea structurării datelor.	+	+				Cu excepția tipului mulțime pentru profilul umanistic.
Mulțimea de valori ale tipurilor de date structurate; Restricțiile impuse de realizările limbajelor de programare în cazul tipurilor de date în studiu.	Exerciții pentru determinarea mulțimii de valori ale tipurilor de date (tablou, șir de caractere, articol, mulțime, fișier).	+	+	Exerciții pentru determinarea restricțiilor impuse mulțimii de valori ale tipurilor de date în studiu pentru diferite realizări ale compilatoarelor limbajelor de programare/în particular – pentru compilatorul utilizat în laboratorul de informatică.	+	+	Exerciții pentru determinarea dimensiunii necesare a variabilelor de tip tablou, șir de caractere, articol, mulțime în funcție de condițiile unei probleme reale propuse.	+	+				Cu excepția tipului mulțime pentru profilul umanistic.

Operațiile destinate prelucrării tipurilor de date structurate.	Exerciții pentru identificarea operațiilor permise pentru datele structurate în studiu. Exerciții pentru determinarea corectitudinii expresiilor care conțin operații/elemente ale structurilor în studiu.	+	+	Exerciții pentru prelucrare a datelor structurate în studiu în conformitate cu setul de operații permise pentru fiecare tip: – operații aritmetice cu elementele structurilor; – operații relaționale cu elementele structurilor; – operații cu mulțimi; – operații cu fișiere text.	+	+	Probleme de prelucrare a datelor structurate în studiu în conformitate cu setul de operații permise pentru fiecare tip: – parcurgerea structurii; – căutarea/inlocuirea elementelor după un criteriu dat; – intersecția, reuniunea, diferența mulțimilor; – deschiderea, închiderea, actualizarea fișierelor text. – excluderea elementelor structurii, după un criteriu dat; – reordonarea elementelor structurii după un criteriu dat.	+	+	Identificarea situațiilor din viața reală și rezolvarea problemelor în care apare necesitatea de integrare a structurilor de date în studiu: – citirea valorilor elementelor structurilor din fișiere text; – prelucrarea tablourilor de șiruri de caractere; – prelucrarea tablourilor de articole.	+	+	Cu excepția operațiilor cu tipul mulțime pentru profilul umanistic. Pentru profilul umanistic problemele de excludere a elementelor din structuri, reordonare a elementelor din structuri pot fi rezolvate în cadrul temei subprograme, cl. XI.
Informația													
Informația: – cantitatea de informație; – stocarea informației;	Exerciții de: – argumentare a necesității determinării cantității de informație;	+	+	Exerciții de: – determinare a cantității de informație în mesajele sursei;	+	+	Elaborarea programelor pentru: – determinarea cantității de informație în mesajele sursei;	+	-	Identificarea situațiilor din viața reală și rezolvarea la calculator a problemelor, în care apare necesitatea de integrare a structurilor de date în studiu	+	-	

– transmiterea informației; – prelucrarea informației.	– identificare a surselor, canalelor și purtătorilor de informație.			– determinare a cantității de informație în texte; – determinare a capacității de stocare a purtătorilor de informații.			– determinare a capacității de stocare a purtătorilor de informații.			determinării cantității de informație, determinării capacității purtătorilor de informație.			
Semne și alfabet. Codificarea și decodificarea mesajelor.	Exerciții de: argumentare a necesității de codificare și decodificare a informației text, audio și video.	+	+	Exerciții de: – codificare și decodificare a informației text;	+	+	Elaborarea programelor pentru: – codificarea și decodificarea informației text; – determinarea cantității de informație în texte.	+	-	Identificarea alfabetelor frecvent utilizate și formularea problemelor informatice în care este necesară codificarea/decodificarea semnelor acestora.	+	-	
Cuantizarea imaginilor. Reprezentarea și transmiterea informației.	Exerciții de: identificare a metodelor de discretizare în spațiu și în valoare a imaginilor statice; – discretizare a imaginilor dinamice în timp.	+	+	– discretizare în spațiu și în valoare a imaginilor statice; – discretizare a imaginilor dinamice în timp; – determinarea cantității de informație în imagini, secvențe audio și video.	+	-	– determinarea cantității de informație în imagini, secvențe audio și video.	+	-				

BAZELE ARITMETICE ALE TEHNICII DE CALCUL											
Sisteme de numerație. Sisteme poziționale și sisteme nepoziționale de numerație. Sistemele poziționale de numerație: – sistemul binar; – sistemul octal; – sistemul hexazecimal. Conversia numerelor din baza 2, 8, 16 în sistemul zecimal și invers: din binar în octal, hexazecimal și invers.	Exerciții de: • argumentare a necesității de aplicare în informatică a aritmeticii de calculator; • identificare a tipului sistemului de numerație.	+	+	Exerciții de: • conversie a numerelor dintr-un sistem de numerație în altul.	+	+	Elaborarea de programe pentru conversia numerelor întregi/reale dintr-un sistem de numerație în altul.	+	-		
Operațiile aritmetice în sistemul binar: – adunarea numerelor binare; – scăderea numerelor binare.	Exerciții de: • argumentare a necesității utilizării în informatică a unor aritmetici de calculator; • utilizare a terminologiei aferente aritmeticii de calculator.	+	-	Exerciții de: • adunare a numerelor binare; • scădere a numerelor binare.	+	-	Elaborarea de programe pentru: • modelarea numerelor binare prin șiruri de caractere/tablouri unidimensionale; • adunarea/scăderea numerelor binare.	+	-		

Reprezentarea numerelor în calculator: – numerele întregi în cod direct, cod invers și cod complementar; – numerele reale în virgulă fixă și virgulă mobilă.	Exerciții de: • explicare a noțiunilor <i>cod direct/invers/complementar</i> ; • identificare a semnului numărului reprezentat în cod direct/invers/complementar.	+	-	Exerciții de: • reprezentare a numerelor în cod direct, cod invers și cod complementar; • reprezentare a numerelor reale în virgulă fixă și virgulă mobilă; • determinare a numărului în sistemul de numerație cu baza 10 după reprezentarea lor în cod direct/invers/complementar;	+	-	Elaborarea de programe pentru reprezentarea numerelor naturale/întregi pe N poziții binare în cod direct, invers, complementar.	+	-	Argumentarea necesității codului invers/complementar pentru reprezentarea numerelor în calculator.	+	-
ALGEBRA BOOLEANĂ (numai profilul real)												
Variabile și expresii logice: – constante și variabile logice; – operatorii logici NU, ȘI, SAU; – expresii logice; – tabelele de adevăr ale expresiilor logice. Funcții logice: – metode de definire a funcțiilor logice; – funcțiile logice frecvent utilizate; – setul complet de funcții logice.	Exerciții de: • utilizare a terminologiei aferente algebrei booleene; • explicare a rolului algebrei booleene în descrierea și proiectarea echipamentelor digitale.	• creare a tabelor de adevăr ale expresiilor logice; • de definire a funcțiilor logice prin tabele și formule; • creare a tabelor de adevăr ale funcțiilor logice.			Probleme de: • creare a tabelor de adevăr ale expresiilor logice cu (și fără)-scrierea programelor într-un limbaj de programare de nivel înalt; • calcul al valorilor funcțiilor logice.							

CIRCUITE LOGICE (numai profilul real)														
Circuite logice elementare. Clasificarea circuitelor logice. Porți logice frecvent utilizate. Circuite combinaționale frecvent utilizate: – sumatorul; – comparatorul; – decodificatorul; – multiplexorul; – demultiplexorul. Circuite secvențiale frecvent utilizate: – bistabilul; – registrul; – numărătorul; – generatoare de impulsuri.	Exerciții de: • descriere a metodelor de reprezentare a valorilor binare prin mărimi fizice; • explicare a impactului pe care îl are modul de reprezentare a variabilelor logice asupra performanțelor echipamentelor de calcul; • clasificare a circuitelor logice.	Exerciții de: • explicare a modului de realizare a funcțiilor logice NU, ȘI, SAU prin circuite cu elemente de comutație; • observare și înțelegere a legăturii între formulele funcțiilor logice și materializarea lor prin reuniunea porților logice;	Exerciții de: • proiectare prin scheme a circuitelor combinaționale frecvent utilizate: sumatorul; comparatorul; • proiectare prin scheme a circuitelor secvențiale frecvent utilizate: bistabilul, registrul, numărătorul.	Exerciții de: • proiectare materială a circuitelor combinaționale frecvent utilizate: sumatorul (1 – 2 biți); comparatorul (1 – 2 biți).										
STRUCTURA CALCULATORULUI ȘI REȚELELE DE CALCULATOARE														
Schema funcțională a calculatorului: – procesorul; – memoria internă; – dispozitivele de intrare-ieșire; – memoria externă.	Exerciții de: • identificare a unităților funcționale ale calculatorului și a traseelor de date între ele.	+	+	• descriere a schemei funcționale a calculatorului.	+	+					Proiect: • evoluția structurii calculatoarelor.	+	-	

Principiul de comandă prin program: – date și instrucțiuni; – formatul instrucțiunilor; – executarea instrucțiunilor.	Exerciții de: • explicare a principiului de comandă prin program; • clasificare a instrucțiunilor în funcție de tipul lor.	+	-	Exerciții de: • modelare a execuției instrucțiunilor în funcție de tipul lor.	+	-							
Resursele calculatorului: – echipamentele; – programele. Memoriile externe: – pe purtători magnetici; – pe purtători optici; – pe semiconductori. Dispozitivele de intrare-ieșire: – vizualizatorul; – tastatura; – șoricelul; – imprimantele. Clasificarea calculatoarelor.	Exerciții de: • evidențiere a resurselor tehnice și a resurselor programate ale calculatorului; • clasificare a calculatoarelor în funcție de caracteristicile tehnico-economice și domeniile de utilizare.	+	+	Exerciții de: • explicare a principiilor de funcționare a memoriilor externe; • explicare a principiilor de funcționare a dispozitivelor de intrare-ieșire.	+	+	* Studii de caz: • stocarea informației pe purtători magnetici și purtători optici; • stocarea informației pe discuri optice și memorii pe semiconductori; • calculatoarele universale și calculatoarele dedicate.	+	-	* Proiecte: • istoria tehnicii de calcul; • evoluția calculatoarelor; • microprocesoarele; • calculatoarele în jurul nostru; • cum să-ți procuri un calculator personal.	+	-	

Tipuri de rețele: – rețele locale; – rețele regionale; – rețele globale. Tehnologii de cooperare în rețea: – de la egal la egal; – client-server. Topologia și arhitectura rețelelor.	Exerciții de: • explicare a principiilor de funcționare a rețelelor de calculatoare; • descriere a conceptului de rețea de calculatoare.	+	+	Exerciții de: • clasificare a rețelelor în funcție de modul de interconexiune a calculatoarelor; • clasificare a rețelelor în funcție de distanța între calculatoare.	+	+	Exerciții de: • observare și înțelegere a legăturii între componentele tehnice și serviciile oferite de rețea; * Studii de caz: • evoluția rețelelor de calculatoare; • evoluția Internetului.	+	-	* Elaborarea proiectelor: • extinderea serviciilor Internet, disponibile în liceu; • extinderea serviciilor Internet, disponibile la domiciliu.	+	-	
Rețeaua Internet. Servicii Internet: – e-comunicarea (e-mail, forum, chat, rețele sociale); – accesul la calculatoarele distanțe; – transferul de fișiere; – localizarea și regăsirea informației.	Exerciții de: • explicare a principiilor de organizare a serviciilor Internet; • explicare a principiilor de comunicare în rețeaua Internet; • explicare a principiilor de căutare și localizare a datelor în rețeaua Internet.	+	+	Exerciții de: • comunicare în rețeaua Internet (e-mail, chat text, audio, video, conversație de grup); • căutare și localizare a datelor în rețeaua Internet.	+	+				Identificarea situațiilor reale în care pot fi utilizate serviciile Internet.	+	+	

Clasa XI

Detalii de conținuturi și activități pe etapele de formare a competențelor după profil

Conținuturi	Fixare cunoștințe	R	U	Funcționalizare	R	U	Interiorizare	R	U	Exteriorizare	R	U	Observații
SUBPROGRAME													
Subprograme: – probleme și subprobleme; – programul principal și programul apelat; – subprograme și apeluri de subprograme; – tipuri de subprograme;	Exerciții de: – folosire a termenilor <i>problemă</i> , <i>subproblemă</i> , <i>program principal</i> , <i>program apelat</i> , <i>subprogram</i> , <i>funcție</i> , <i>procedură</i> ; – explicare a modului de execuție a apelurilor de funcții și proceduri; – argumentare a necesității divizării problemelor complexe în subprobleme mai simple; – definire a subprogramelor prin formule metalingvistice și diagrame sintactice; – identificare a subproblemelor, a căror soluționare necesită utilizarea subprogramelor.	+	+	Exerciții de: – utilizare a funcțiilor și procedurilor predefinite ale limbajului; – elaborare a programelor care utilizează funcțiile predefinite.	+	+	Probleme de – elaborare a subprogramelor definite de utilizator: <i>Suma valorilor elementelor unui șir de numere, media aritmetică, geometrică; cel mai mare divizor comun, cel mai mic multiplu comun a două numere; sortarea unui șir de numere;</i>	+	+	Probleme de – elaborare a programelor cu subprogram, care modelează situații și fenomene din lumea reală: <i>calculul ratingului elevilor din clasă după reușită, determinarea liderului după rating; validarea dreptului de acces la program (verificare nume utilizator/parolă) etc.;</i>	+	+	
							– elaborare a subprogramelor definite de utilizator pentru prelucrarea textelor: <i>căutarea secvenței de caractere în text, înlocuirea, excluderea, modelarea dicționarului bilingv etc.</i>			– elaborare a programelor pentru prelucrarea textelor din fișiere text cu subprogram de: <i>scriere a textului în fișier extern, citire a textului din fișierul extern, căutarea, înlocuirea excluderea, secvențelor de caractere etc.</i>			Problemele din acest compartiment sînt opționale pentru profilul real.

Comunicarea între programul/sub-programul apelant și subprogramul apelat: – modul de transfer al controlului în cazul apelului de subprograme; – modul de transmitere a argumentelor și de returnare a rezultatelor. Domenii de vizibilitate: – structura de bloc a programelor; – variabile globale și variabile locale.	Exerciții de: – observare și înțelegere a legăturilor informaționale (date) și de control (apeluri) între programul principal și subprogramele apelate; – înțelegere a termenilor parametru formal/actual, parametru valoare/variabilă; – delimitare a zonei de vizibilitate pentru variabilele declarate.	+	+	Exerciții de: – transfer al datelor între programul principal și subprogram; – control al valorilor variabilelor globale și locale.	+	+	Probleme de analiză (modificare) a structurilor de date: <i>numărul de elemente cu valori pozitive/într-un tablou bidimensional, suma, produsul elementelor dintr-o zonă a tabloului</i> etc.	+	+	Probleme de elaborare a programelor cu subprograme, care modelează situații și fenomene din lumea reală: <i>determinarea zonelor unei regiuni, care vor fi inundate la creșterea nivelului apei cu o valoare dată, determinarea culmilor și a coordonatelor lor într-o zonă deluroasă</i> etc.	+	+	
Efectele colaterale: – cauzele efectelor colaterale; – modul de evitare a efectelor colaterale; – elaborarea de programe în care nu apar efecte colaterale.	Exerciții de: – înțelegere a noțiunii de efect colateral; – observare a efectelor colaterale în program.	+	-	Exerciții de: – determinare a erorilor, generate de efectele colaterale; – eliminare a efectelor colaterale din program.	+	-	Elaborare a programelor în care nu apar efecte colaterale.	+	-				

Recursia: – modul de transmitere a controlului la execuția algoritmilor recursivi; – modul de alocare a memoriei la execuția algoritmilor recursivi; – avantajele și neajunsurile recursiei; – elaborarea de subprograme recursive.	Exerciții de: – înțelegere a noțiunii de recursie; – observare a proceselor recursive în natură și mediul social; – identificare a condițiilor care determină un proces recursiv; – determinare a cazurilor elementare și a formulelor de reducere; – identificare a zonei de memorie, în care sînt stocate datele la apeluri recursive; – identificarea principiilor de rezervare a memoriei în cazul apelurilor recursive.	+	-	Exerciții de: – modelare a procesului de calcul recursiv; – urmărire a cantității necesare de memorie la apelurile recursive; – elaborare a subprogramelor recursive elementare: (<i>calculul puterii, factorialului, termenilor șirului Fibonacci</i> etc.); – elaborare paralelă a subprogramelor recursive și iterative pentru aceeași problemă și compararea avantajelor și neajunsurilor fiecăreia.	+	-	Probleme de – elaborare a programelor cu subprograme recursive: <i>Funcția Ackerman, polinoamele Cebishev, polinoamele Hermit, combinații ale elementelor</i> etc.	+	-	Probleme de elaborare a programelor cu subprograme, recursive care modelează situații și fenomene din lumea reală: <i>determinarea zonelor negre/numărului de obiecte pe o fotografie, construirea structurilor fractale, calculul valorii unei expresii</i> etc.	+	-	
Sintaxa declarațiilor și apelurilor de subprograme.	Exerciții de – identificare a elementelor de structură a subprogramelor; – verificare a sintaxei declarațiilor în conformitate cu diagramele sintactice și formulele metalingvistice.	+	+	Exerciții de – excludere a erorilor de sintaxă în declarațiile și apelurile de subprograme.	+	+							

STRUCTURI DINAMICE DE DATE (profil real)					
Variabile dinamice. Tipul referință: – variabile statice; – variabile dinamice; – referirea variabilelor dinamice; – tipul referință; – alocarea statică și alocarea dinamică a memoriei; – crearea și de distrugerea variabilelor dinamice.	Exerciții de: – introducere intuitivă (prin desen) a metodelor de alocare dinamică a memoriei; – argumentare a necesităților de utilizare a variabilelor dinamice.	Exerciții de: – elaborare a programelor în care se utilizează variabile dinamice; – explicare a modului de alocare a memoriei operative în cazul utilizării variabilelor statice și variabilelor dinamice; – crearea variabilelor dinamice; – distrugerea variabilelor dinamice.	Probleme a căror rezolvare necesită alocarea dinamică a memoriei: – compararea/adunarea/scăderea perechilor de elemente din tablouri unidimensionale de dimensiuni mari (ex 30000 elemente cu valori întregi fiecare); – prelucrarea concomitentă a tablourilor bidimensionale de dimensiuni mari.		
Structuri de date: – structuri implicite și structuri explicite; – structuri statice și structuri dinamice; – structuri omogene și structuri eterogene; – structuri recursive.	Exerciții de: – evidențiere a diferențelor dintre structurile implicite și structurile explicite de date, dintre structurile omogene și structurile eterogene de date, dintre structurile statice și structurile dinamice de date;				

Structuri dinamice de date: liste unidirecționale, stive, cozi; crearea structurilor în studiu; prelucrarea structurilor în studiu; extragerea datelor din structurile în studiu.	– selecție a problemelor, a căror soluționare necesită utilizarea structurilor de date propuse în studiu.	Exerciții de: – explicare a modului de alocare a memoriei operative în cazul utilizării structurilor dinamice de date; – stocare și de prelucrare a datelor cu ajutorul listelor, stivelor, cozilor; – realizare a subprogramelor pentru efectuarea operațiilor asupra structurilor de date: <i>inițializare, adăugare element, excludere element, parcurgere structură.</i>	*Elaborarea programelor de gestionare a structurilor de date în studiu: – inițializare; – adăugare a unui element; – excludere a unui element; – parcurgere a structurii; – căutare a unui element după o cheie dată.	* Probleme a căror rezolvare necesită utilizarea structurilor dinamice de date în studiu: – gestionarea tranzacțiilor la bancă; – operații cu polinoame; – gestionarea garniturilor de tren în depou.	
Arbori binari: crearea arborilor binari; parcurgerea arborilor binari; modificarea structurii arborilor binari.	Exerciții de: – selecție a problemelor a căror soluționare necesită utilizarea structurilor de date în studiu; – descriere intuitivă (prin desen) a metodelor de parcurgere a arborilor binari (<i>inordine, preordine, postordine</i>); – determinare a înălțimii arborilor binari; – determinare a nodurilor amplasate la un nivel dat.	Exerciții de: – organizare și de prelucrare a datelor cu ajutorul arborilor binari; – descriere intuitivă (prin desen) a operațiilor de adăugare și excludere a elementelor în arborii binari.	*Elaborarea programelor de gestionare a arborilor binari: – inițializare; – adăugare a unui nod; – parcurgere a arborelui; – căutare a unui nod după o cheie dată; – excludere a unui nod după o cheie dată.	* Probleme a căror rezolvare necesită utilizarea arborilor binari arbitrari sau cu o structură predefinită: – crearea arborelui binar asociat unei expresii aritmetice și evaluarea expresiei; – organizarea căutărilor multiple eficiente a elementelor.	

Tipul de date pointer: – mulțimea de valori ale tipului pointer; – mulțimea de operații ale tipului pointer. Gestionarea memoriei interne a calculatorului.	Exerciții de: – descriere a mulțimii de valori a tipului pointer; – descriere a mulțimii de operații a tipului pointer; – descriere a principiilor de gestionare a memoriei interne a calculatorului.	Exerciții de: – utilizare a instrumentelor standard de alocare a memoriei pentru structuri de date; – utilizare a instrumentelor standard de eliberare a memoriei; – utilizare a instrumentelor standard pentru estimarea volumului de memorie disponibilă.			
TEHNICI DE PROGRAMARE (profil real)					
Analiza algoritmilor: – estimarea necesarului de memorie; – complexitatea temporală a algoritmului; – clasificarea algoritmilor în funcție de complexitate.	Exerciții de: – utilizare a terminologiei aferente complexității algoritmilor; – argumentare a necesităților de analiză a complexității algoritmilor; – identificare a tipului de complexitate al algoritmului.	Exerciții de – estimare a necesarului de memorie; – estimare a numărului de operații; – estimare a necesarului de memorie; – calcul direct al timpului de execuție al programului/secvenței de program.	Experiment: Determinarea directă a timpului necesar pentru algoritmul de sortare prin metoda bulelor a elementelor unui tablou unidimensional cu N elemente de tip întreg pentru N=1000; N=5000; N=10000; N=20000.	* Studiu de caz: determinarea complexității algoritmului de generare a tuturor submulțimilor unei mulțimi din N elemente.	
Abordări iterative și recursive.	Exerciții de: – explicare a noțiunilor aferente fiecărui tip de abordare; – identificare a tipului de abordare;	Exerciții de – estimare a dimensiunii limită a datelor de intrare în cazul abordării recursive; – estimare a dimensiunii limită a datelor de intrare în cazul abordării iterative;	Elaborarea programelor pentru calculul iterativ/recursiv al – termenilor șirului Fibonacci; – combinații C_n^k; – aranjamente A_n^k.	* Proiect: – analiza comparativă a complexității temporale pentru abordările recursive și iterative ale problemelor combinatorice;	

	– descriere a avantajelor și neajunsurilor fiecărui tip de abordare.	– estimare a complexității temporale în cazul abordării recursive; – estimare a a complexității temporale în cazul abordării iterative.		– analiza comparativă a dimensiunilor limită a datelor de intrare pentru abordările recursive și iterative ale problemelor combinatorice.	
Metoda trierii: – tehnica de triere; – complexitatea metodei de triere; – domeniile de aplicare a metodei de triere.	Exerciții de: – descriere intuitivă a mecanismului metodei; – identificare a claselor de probleme, pentru care poate fi folosită metoda trierii.	Exerciții de: – estimare a complexității temporale a metodei; – algoritmizare a mecanismului general al metodei.	Elaborarea programelor pentru problemele clasice și problemele reductibile la ele: – <i>problema generării secvențelor binare de o lungime dată;</i> – <i>problema generării tuturor submulțimilor unei mulțimi;</i> – <i>problema generării tuturor partițiilor unei mulțimi;</i> – <i>probleme de triere în plan (apartență la un domeniu).</i>	Identificarea problemelor din viața reală care pot fi rezolvate folosind metoda trierii și elaborarea programelor pentru rezolvarea lor. Ex: – <i>problema determinării codului de acces;</i> – <i>problema rucsacului;</i> – <i>determinarea mulțimii maxime independente etc.</i>	
Metoda Greedy: – tehnica Greedy; – complexitatea metodei Greedy; – domeniile de aplicare a metodei Greedy.	Exerciții de: – descriere intuitivă a mecanismului metodei; – descriere a rolului algoritmilor de sortare folosiți.	Exerciții de: – estimare a complexității temporale a metodei; – algoritmizare a mecanismului general al metodei; – stabilire optimă a criteriilor pentru includerea elementelor în soluție.	Elaborarea programelor pentru problemele clasice și problemele reductibile la ele: – <i>problema spectacolelor;</i> – <i>problema rucsacului (și variațiile lor).</i>	Identificarea problemelor din viața reală care pot fi rezolvate optim folosind metoda Greedy și elaborarea programelor pentru rezolvarea lor. Ex: – <i>problema arborelui de cost minim;</i> – <i>plata sumei cu un număr minim de bancnote;</i> – <i>formarea pereților de separare de rezistență maximă etc.</i>	

				*Proiect: Studiul comparativ al soluțiilor obținute prin metoda Greedy și metoda trierii pentru: <i>problema rucsacului, problema spectacolelor etc.</i>	
Metoda reluării: – tehnica de reluare; – complexitatea metodei reluării; – domeniile de aplicare a metodei reluării.	Exerciții de: – descriere intuitivă a mecanismului metodei; – descriere a rolului abordării recursive pentru metodă; – reprezentare grafică a proceselor de transfer a controlului/pentru abordarea recursivă.	Exerciții de: – estimare a complexității temporale a metodei; – algoritmizare a mecanismului general al metodei.	Elaborarea programelor pentru problemele clasice și problemele reductibile la ele: – <i>descompunerile numărului în sumă de termeni naturali;</i> – <i>problema damelor;</i> – <i>Atilla și regele;</i> – <i>acoperirea tablei de șah cu mișcări ale calului etc. (și variațiile lor).</i>	Identificarea problemelor din viața reală care pot fi rezolvate folosind metoda reluării și elaborarea programelor pentru rezolvarea lor. Ex: – problema comis-voiajorului; – căutarea drumurilor în labirint. *Proiect: Studiul comparativ al complexității implementării iterative a metodei reluării și a implementării recursive.	
* Metoda desparte și stăpânește: tehnica desparte și stăpânește; complexitatea metodei desparte și stăpânește; domeniile de aplicare a metodei desparte și stăpânește.	Exerciții de: – descriere intuitivă a mecanismului metodei; – descriere a rolului abordării recursive pentru metodă; – reprezentare grafică a proceselor de transfer a controlului/pentru abordarea recursivă.	Exerciții de: – estimare a complexității temporale a metodei; – algoritmizare a mecanismului general al metodei.	Elaborarea programelor pentru problemele clasice și problemele reductibile la ele: – <i>turnurile din Hanoi;</i> – <i>sortarea rapidă;</i> – <i>sortarea prin interclasare;</i> – <i>căutarea binară.</i>	Proiect: Studiul comparativ al eficienței algoritmilor de sortare.	

Clasa XII

Detalii de conținuturi și activități pe etapele de formare a competențelor după profil

Conținuturi	Fixare cunoștințe	R	U	Funcționalizare	R	U	Interiorizare	R	U	Exteriorizare	R	U	Observații
Elemente de modelare (numai profilul real)													
Model și modelare. Clasificarea modelelor: – materiale; – ideale; – matematice. Modelare matematică.	Exerciții de: – identificare și clasificare a modelelor; – explicare a sensului termenilor model și modelare; – enumerare a criteriilor de clasificare a modelelor.			Exerciții de: – selectare a caracteristicilor obiectului modelat în funcție de destinația modelului; – elaborare a modelelor ideale pentru fenomene fizice; – elaborare a modelelor ideale pentru corpuri geometrice bidimensionale și tridimensionale; – elaborare a modelelor matematice pentru fenomene și situații de natură fizică, biologică, chimică, socială.			Probleme de: – elaborare a modelelor matematice pentru probleme din viața reală; – transpunere a modelelor matematice elaborate într-un limbaj de programare de nivel înalt.			Probleme de cercetare a rezultatelor modelării în funcție de valorile datelor inițiale. Studii de caz: Aplicațiile modelării matematice în cinematografia modernă/arhitectură/design de interior etc.			
Soluții analitice și soluții de simulare.	Exerciții de: – diferențiere a soluțiilor analitice și soluțiilor de simulare; – identificare a problemelor care pot fi soluționate prin metode analitice și a problemelor care pot fi soluționate prin metode de simulare;			Exerciții de: – determinare a formulelor pentru soluțiile analitice; – determinare a formulelor pentru soluțiile de simulare.			Probleme de identificare a proceselor, fenomenelor vieții reale, controlul evoluției cărora poate fi realizat prin modelarea soluțiilor de simulare.			Probleme de automatizare a controlului prin elaborarea programelor de modelare într-un limbaj de programare de nivel înalt (controlul coordonatelor curente ale unui robot, controlul nivelului lichidului dintr-un bazin etc.).			

	– explicare a metodelor generale de obținere a soluțiilor analitice și a soluțiilor de simulare.				
Etapele rezolvării problemelor la calculator.	Exerciții de: – identificare a etapelor rezolvării problemelor la calculator; – diferențiere a etapelor rezolvării problemelor la calculator;	Exerciții de explicare a interacțiunii între: model matematic și algoritm, algoritm și program, resurse program și resurse calculator; planificare a procesului de rezolvare a problemei pe calculator.	Rezolvarea pe etape a problemelor algoritmice standard (determinarea elementelor cu valoare maximă/minimă în tablou, rezolvarea ecuațiilor de gradul doi etc.).	Identificarea problemelor din viața reală care pot fi rezolvate urmînd etapele rezolvării problemei la calculator și determinarea soluțiilor acestora.	
CALCUL NUMERIC (numai profilul real)					
Erori în calculele numerice: eroarea absolută; eroarea relativă. Erori ale datelor de intrare; de rotunjire; de aproximare; * de metodă; * de problemă.	Exerciții de: – identificare a soluției exacte și a soluției calculate; – identificare a erorilor datelor de intrare, de rotunjire, de aproximare.	Exerciții de calcul al erorii absolute și al erorii relative: – pentru măsurări simple; – pentru soluții ale problemelor obținute prin metoda Greedy (problema rucsacului etc.).	Probleme de: – modelare a calculatorului aritmetic standard (extins) într-un limbaj de programare de nivel înalt; – estimare a erorii valorilor calculate cu ajutorul programelor de calculator în funcție de tipul folosit pentru datele din program; – evaluare a erorilor de aproximare, de rotunjire; *- evaluare a erorilor de problemă, de metodă.	* Proiect: Elaborarea într-un limbaj de nivel înalt a unui program de adunare (scădere) exactă a numerelor întregi mari (pînă la 100 de cifre).	

Rezolvarea pe calculator a ecuațiilor algebrice și transcendente: – soluția ecuației; – separarea soluțiilor; – metoda bisecției; – metoda coardelor; – metoda lui Newton; – formule recurente; – extremități fixe; – aproximare inițială.	Exerciții de – separare a soluțiilor ecuațiilor prin metoda tririi; * – separare a soluțiilor ecuațiilor cu ajutorul aplicațiilor de trasa-re a graficelor; – descriere a algoritmului de reali-zare a metodelor în studiu.	Exerciții de – verificare a condițiilor de aplicare a metodelor în studiu pentru ecuațiile propuse; – determinare a extremității fixe în metoda coar-delor; – selectare a apro-ximării inițiale în metoda lui Newton.	Probleme de elaborare a progra-melor care realizează calculul iterativ al soluțiilor ecuațiilor prin metodele bisecției, coardei, Newton.	Identificarea și rezolvarea numerică a problemelor din viața reală al căror model matematic este descris prin ecuații algebrice și trans-cendente. * Combinarea metodei coardelor și a metodei Newton pentru optima-rea determinării soluției ecuației.	
Calculul determinanți-lor algoritmi recursivi; algoritmi iterativi. Regula lui Cramer. * Metoda lui Gauss.	Exerciții de – descriere în limbaj natural sau pseudocod a algoritmului de calcul al de-terminanților numerici; – estimare a numărului de operații necesare pentru a calcula determinanții; – descriere a algo-ritmilor de rezol-vare a sistemelor de ecuații liniare pentru metodele în studiu.	Exerciții de – programare a algoritmilor de calcul numeric al determinan-ților; – prelucrare a tablourilor bidi-mensionale; – estimare a numărului de operații necesare pentru a rezol-va sistemul de ecuații liniare.	Rezolvare de probleme: – rezolvarea problemelor, mo-delul matematic al cărora se reduce la calculul determi-nanților numerici; – integrare a calculului de-terminanților numerici în programarea algoritmilor de rezolvare a sistemelor de ecuații liniare.	Rezolvare de probleme al căror model matematic reprezintă sisteme de ecua-ții liniare. *Experiment: determinarea dimensiunii maxime a datelor de intrare pentru care este aplicabilă abordarea recursivă a cal-culului determinantului. *Studiu de caz: analiza comparativă a valorilor determinanților, obținute prin algoritmi recursivi și iterativi.	

Calculul numeric al integralelor. Metoda dreptunghiurilor: – de mijloc; – de stînga; – de dreapta. * Metoda trapezelor.	Exerciții de: descriere a metodei dreptunghiurilor (și a variațiilor ei) pentru calculul integralei definite; * descriere a metodei trapezelor pentru calculul integralei definite.	Exerciții de: – programare a algoritmului pentru calculul numeric al integralelor prin metoda dreptunghiurilor și a variațiilor ei; * – programare a algoritmului pentru calculul numeric al integralelor prin metoda trapezelor.	Probleme de – calcul numeric al ariilor figurilor curbilinii plane prin metoda dreptunghiurilor și a variațiilor ei; * – calcul numeric al ariilor figurilor curbilinii plane prin metoda trapezelor.	Rezolvare de probleme al căror model matematic necesită calculul integralei definite. *Studiu de caz: analiza comparativă a valorii calculate a integralei definite realizate prin metodele dreptunghiurilor și a valorii calculate analitic sau cu ajutorul aplicațiilor online de calcul al integralei definite.	
BAZE DE DATE					
Noțiuni și concepte. Tipuri de baze de date: – ierarhice; – în rețea; – relaționale.	Exerciții de: – explicarea sensului termenilor <i>bază de date, sistem de gestiune a bazelor de date</i> ; – introducerea intuitivă (prin desen) a structurii bazelor de date ierarhice, în rețea și relaționale; – diferențiere a termenilor în studiu.	+	+		

Sisteme de gestiune a bazelor de date: – structura; – funcțiile. Etapele de elaborare a unei baze de date.	Exerciții de: – explicare a destinației fiecărui obiect al bazei relaționale de date; – diferențiere a etapelor de elaborare a unei baze de date și explicare a conținutului fiecărei etape; – explicarea rolului persoanelor antrenate în elaborarea și utilizarea bazelor de date.	+	+	Exerciții de: – reprezentare prin desen a structurii sistemelor de gestiune a bazelor de date; – reprezentare prin desen a traficului de date între obiectele unei baze relaționale de date.	+	+	Selectarea temei proiectului colectiv (de clasă) pentru dezvoltarea ulterioară a bazei de date. Proiectarea structurii bazei de date în conformitate cu tema selectată.	+	+	Selectarea/reparțizarea temelor proiectelor individuale (sau de grup) pentru dezvoltarea ulterioară. Proiectarea de către elevi/grupuri a structurii bazei de date în conformitate cu tema selectată.	+	+	
Tabele: – destinația și structura tabelor; – crearea tabelor; – introducerea datelor în tabel; – modificarea structurii unui tabel; – editarea înregistrărilor; – sortarea înregistrărilor; – căutarea și înlocuirea valorilor;	Exerciții de: – diferențiere a tipurilor de date în câmpurile unui tabel; – definire a proprietăților câmpurilor; – alegere a cheii primare; – explicare a metodelor de introducere a datelor în tabele;	+	+	Exerciții de: – creare a tabelor cu ajutorul sistemului de asistență sau prin proiectare independentă; – introducere a datelor în tabelele create; – modificare a structurii tabelor;	+	+	Crearea tabelor pentru baza de date (proiectul de clasă); Introducerea datelor în tabelele bazei de date (proiectul de clasă); Redactarea datelor în tabelele bazei de date (proiectul de clasă); Stabilirea corelațiilor între tabelele bazei de date (proiectul de clasă).	+	+	Crearea tabelor pentru baza de date selectată; Introducerea datelor în tabelele bazei de date selectate; Redactarea datelor în tabelele bazei de date selectate; Stabilirea corelațiilor între tabelele bazei de date selectate. *Studii de caz: – noțiuni generale despre normalizarea unui tabel;	+	+	

- crearea și folosirea unui filtru; - stabilirea corelațiilor între tabele.	- introducere intuitivă (prin diagrame de corelare) a noțiunii de tabele corelate; - explicare a proprietăților fiecărui tip de corelație.			- editare a înregistrărilor; - formatare a datelor; - stabilire a corelațiilor între tabele; - sortare a înregistrărilor, - căutare și înlocuire a valorilor; - elaborare a filtrelor;				- validarea datelor introduse într-un câmp al tabelului; - utilizarea măștilor; - asigurarea integrității datelor; - editarea în cascadă a datelor.					
Interogări: - destinația și structura interogărilor; - crearea unei interogări; - sortarea și gruparea înregistrărilor. Expresii: - construirea expresiilor; - interogări pentru adăugare, actualizare și eliminare; - gruparea și totalizarea datelor într-o interogare.	Exerciții de: - introducere intuitivă (prin desen) a noțiunilor interogare și set dinamic rezultat al interogării; - scriere și de evaluare a expresiilor; - construire a expresiilor cu ajutorul sistemului de asistență.	+	+	Exerciții de: - creare a interogărilor predefinite cu ajutorul sistemului de asistență; - creare a interogărilor simple și complexe; - elaborare a interogărilor în studiu; - creare și modificare a interogărilor de acțiune;	+	+	Formularea cererilor de interogare pentru baza de date (proiectul colectiv); Crearea interogărilor pentru baza de date (proiectul colectiv); Sortarea și gruparea înregistrărilor bazei de date (proiectul colectiv); Formularea cererilor de interogare pe baza expresiilor și testarea lor pentru bază de date (proiectul colectiv); Adăugarea, actualizarea și eliminarea înregistrărilor bazei de date (proiectul colectiv);	+	+	Formularea cererilor de interogare pentru baza de date (individual); Crearea interogărilor pentru baza de date (individual); Sortarea și gruparea înregistrărilor bazei de date (individual); Formularea cererilor de interogare pe baza expresiilor și testarea lor pentru baza de date selectată (individual); Adăugarea, actualizarea și eliminarea înregistrărilor bazei de date (individual);	+	+	

				– grupare și totalizare a datelor în interogările propuse de profesor sau elaborate de elevi.			Gruparea și totalizarea datelor bazei de date (proiectul colectiv).			Gruparea și totalizarea datelor bazei de date (individual).			
Formulare: – destinația și structura formularelor; – crearea și funcționarea unui formular; – modificarea formularelor; – folosirea formularelor; – crearea formularelor pe baza tabelelor corelate; – formatarea datelor.	Exerciții de: – reprezentare intuitivă (prin desen) a structurii formularelor și a fluxului de date între formulare și celelalte obiecte ale bazei de date; – stabilire a proprietăților formularelor, controalelor și secțiunilor; – particularizare a formularelor; – folosire a formularelor; – explicare a modului de funcționare a formularelor pe baza tabelelor corelate; – proiectare a interogărilor pe tabele corelate ca bază pentru formulare.	+	+	Exerciții de: – crearea formularelor predefinite cu ajutorul sistemului de asistență; – modificare a formularelor; – formatarea datelor din baze de date; – crearea și folosirea subformulelor.	+	+	Crearea formularelor pe baza unui tabel a bazei de date (proiectul colectiv); Folosirea și modificarea formularelor bazei de date (proiectul colectiv); Crearea formularelor pe baza tabelelor corelate a bazei de date (proiectul colectiv); Formatarea datelor bazei de date (proiectul colectiv).	+	+	Crearea formularelor pe baza unui tabel a bazei de date (proiect individual); Folosirea și modificarea formularelor bazei de date (proiect individual); Crearea formularelor pe baza tabelelor corelate a bazei de date (proiect individual); Formatarea datelor bazei de date (proiect individual).	+	+	

Rapoarte: – destinația și structura rapoartelor; – crearea și funcționarea unui raport; – crearea rapoartelor pe baza tabelelor corelate; – gruparea și totalizarea datelor într-un raport.	Exerciții de: – reprezentare intuitivă (prin desen) a structurii rapoartelor și a fluxului de date între raport și celelalte obiecte ale bazei de date; – analiză de structură și de particularizare a raportului; – grupare și totalizare a datelor în rapoarte.	+	+	Exerciții de: – elaborare a rapoartelor cu ajutorul sistemului de asistență; – creare și folosire a rapoartelor și subrapoartelor pe baze de date propuse de profesor sau elaborate de elevi; – compactare și reparare a unei baze de date; – creare a parolelor de acces pentru diferite tipuri de utilizatori ai bazei de date.	+	-	Crearea rapoartelor pe baza unui tabel a bazei de date (proiectul colectiv); Crearea rapoartelor pe baza tabelelor corelate a bazei de date (proiectul colectiv); Gruparea și totalizarea datelor într-un raport a unei baze de date (proiectul colectiv); * Compactarea și repararea bazei de date (proiectul colectiv); * Crearea parolelor de acces pentru diferite tipuri de utilizatori ai bazei de date (proiectul colectiv).	+	+	Crearea rapoartelor pe baza unui tabel a bazei de date (proiect individual); Crearea rapoartelor pe baza tabelelor corelate a bazei de date (proiect individual); Gruparea și totalizarea datelor într-un raport a unei baze de date (proiect individual); * Compactarea și repararea bazei de date (proiect individual); * Crearea parolelor de acces pentru diferite tipuri de utilizatori ai bazei de date (proiect individual).	+	+	
---	--	---	---	---	---	---	--	---	---	--	---	---	--

ELEMENTE DE WEB DESIGN												
Documente Web: – noțiuni și concepte; – formatele documentelor Web; – structura unui site; Documente Web: – cerințe; – etape de elaborare; – resurse soft pentru creare și generare.	Exerciții de: – explicare a sensului termenilor document Web, site; – diferențiere a etapelor de elaborare a unui document Web și de explicare a conținutului fiecărei etape.	+	+	Exerciții de: – reprezentare prin desen a structurii unui site.	+	+	Selectarea temei unui site pentru dezvoltare ulterioară (proiect colectiv); Formularea cerințelor către proiectul selectat; Elaborarea structurii pentru proiectul selectat.	+	+	Selectarea temei unui site pentru dezvoltare ulterioară (proiect individual/de grup); Formularea cerințelor către proiectul selectat; Elaborarea structurii pentru proiectul selectat.	+	+
Formatul HTML. Crearea documentelor HTML cu ajutorul aplicațiilor de oficiu. Publicarea documentelor Web în Internet.	Exerciții de: – identificare a metodelor de creare a documentelor HTML cu ajutorul aplicațiilor de oficiu.	+	+	Exerciții de: – creare a documentelor HTML cu ajutorul aplicațiilor de oficiu; – *publicarea documentelor Web în Internet.	+	+	* Studiu de caz „Publicarea pe Web a documentelor HTML”.	+	+			

Structura generală a unui document HTML. Formatarea textului: – titlu; – paragrafe; – comentarii; – stiluri fizice; – stiluri logice; – linii orizontale. Liste: – ordonate; – neordonate; – de definiții. Tipuri de marcare. Tipuri de numerotare. Liste imbricate.	Exerciții de: – identificare a elementelor de structură a unui fișier HTML; – diferențiere a metodelor de evidențiere a textului; – selectare a stilului pentru evidențierea textului; – identificare a listelor într-un cod HTML; – diferențiere a metodelor de marcare/numerotare în liste.	+	+	Exerciții de: – creare a unui document HTML simplu (fără legături, care afișează doar texte și linii orizontale); – formatare a textului cu ajutorul instrumentelor HTML; – creare a listelor (ordonate, neordonate, de definiții); – formatare a listelor (ordonate, neordonate, de definiții); – creare a listelor imbricate.	+	+	Crearea documentelor HTML independente – componente ale proiectului colectiv selectat; Completarea cu informații text a documentelor HTML ale proiectului colectiv; Formatarea textului în documentele HTML ale proiectului colectiv; Inserarea listelor în documentele HTML ale proiectului colectiv.	+	+	Crearea documentelor HTML independente – componente ale proiectului individual/de grup; Completarea cu informații text a documentelor HTML ale proiectului individual/de grup; Formatarea textului în documentele HTML ale proiectului individual/de grup; Inserarea listelor în documentele HTML ale proiectului individual/de grup.	+	+	
---	--	---	---	--	---	---	--	---	---	---	---	---	--

Legături externe, interne: – referință; – cale; – anchoră; – comentariu la legătură. Imagini: – dimensiuni; – chenar; – legături; – comentarii. Tabele: – titlu; – linie; – coloană; – celulă; – contur.	Exerciții de: – identificare a legăturilor către un document extern/ către un site/ către o secvență a aceluiasi sau a altui document; * – identificare a legăturilor către un fișier format arbitrar cu scopul creării unei copii pe disc a acestui fișier; – recunoaștere a formatelor de fișiere-imagini; – recunoaștere a posibilităților de control al imaginilor; – identificare a tabelelor în documente HTML; – recunoaștere a posibilităților de control al tabelelor.	+	+	Exerciții de: – creare a legăturilor către un document extern (aflat în același sau în alt catalog); – creare a legăturilor către un site; – creare a legăturilor către o secvență a aceluiasi sau a altui document; * – creare a legăturilor către un fișier format arbitrar cu scopul creării unei copii pe disc a acestui fișier; – inserare în documente HTML a imaginilor; – creare a tabelelor în documente HTML; – utilizare a tabelelor pentru efecte de design ale documentelor HTML.	+	+	Stabilirea legăturilor între documentele HTML – componente ale proiectului colectiv selectat; Inserarea imaginilor în documentele HTML ale proiectului colectiv; Inserarea tabelelor în documentele HTML ale proiectului colectiv; Formatarea documentelor HTML ale proiectului colectiv cu ajutorul tabelelor; Aplicarea cu ajutorul tabelelor a elementelor de design documentelor HTML ale proiectului colectiv.	+	+	Stabilirea legăturilor între documentele HTML – componente ale proiectului individual/de grup; Inserarea imaginilor în documentele HTML ale proiectului individual/de grup; Inserarea tabelelor în documentele HTML ale proiectului individual/de grup; Formatarea documentelor HTML ale proiectului individual/de grup cu ajutorul tabelelor; Aplicarea cu ajutorul tabelelor a elementelor de design documentelor HTML ale proiectului individual/de grup.	+	+	
---	--	---	---	--	---	---	--	---	---	---	---	---	--

9. Recomandări metodice: proiectarea didactică pe profiluri și clase

Curriculumul la disciplina Informatica, în calitate de parte componentă a curriculumului național, reprezintă un document școlar de tip reglator fiind și un instrument de lucru al profesorului. Proiectarea demersului didactic, care pregătește desfășurarea activității didactice din clasă, presupune parcurgerea de către profesor a următorilor pași:

A. în mod tradițional: lectura curriculumului la disciplină, elaborarea proiectării de lungă durată/în bază de lecții, elaborarea proiectelor didactice ale lecțiilor separate;

B. în contextul instruirii bazate pe competențe: lectura curriculumului la disciplină; identificarea unităților de învățare care stau la baza planificării calendaristice; elaborarea proiectării de lungă durată în baza unităților de învățare; proiectarea separată a unităților de învățare cu identificarea în cadrul acestora a lecțiilor separate.

Abordările diferă prin structurile care se proiectează, atât la etapa inițială (de lungă durată), cit și la etapa finală (proiect didactic: *lecție (A)* și *unitate de învățare (B)*).

Definițiile de lucru ale acestor noțiuni indică univoc orientarea către realizarea unor obiective (lecția) sau dezvoltării unor competențe (unitatea de învățare).

Deși proiectarea didactică în baza unităților de învățare este mai adecvată instruirii orientate către formarea competențelor, cadrele didactice pot folosi în continuare ambele modele de proiectare, precum și variații ale lor.

Proiectarea de lungă durată (cu proiecte didactice pe lecții) poate fi ușor adaptată pentru utilizare în contextul curriculumului modernizat. Elementul pivot al proiectării tradiționale este coloana conținuturilor didactice detaliate, în funcție de care se stabilesc subcompetențele și competențele specifice corelate. Astfel modelul tradițional al proiectării de lungă durată poate fi ajustat la cerințele curriculumului modernizat prin adăugarea a două coloane, care vor conține sub-competențele vizate de conținuturile enumerate și competențele specifice corespunzătoare (tabelul 3).

Metodologia elaborării proiectării de lungă durată în baza unităților de învățare este relativ nouă. Din acest motiv vom prezenta algoritmul de elaborare a proiectării de lungă durată în baza unităților de învățare:

Pasul 1: Lectura personalizată a programei școlare. Libertatea de acțiune a profesorului presupune selectarea pe parcursul lecturii a ordinii de parcurgere a temelor/conținuturilor, a alocărilor de timp, a activităților de învățare și de evaluare.

Pasul 2: Identificarea unităților de învățare:

- Identificăm teme majore ale programei;
- Identificăm conținuturi din curriculum care pot fi asociate unei anumite teme;
- Stabilim competențele specifice/subcompetențele corelate la conținuturi;
- Detaliem conținuturile după criteriul relevanței în raport cu competențele vizate;
- Verificăm în ce măsură ansamblul competențe – conținuturi permite o evaluare pertinentă; eventual, renunțăm la unele conținuturi, pe care le vom avea în vedere pentru altă/alte unități de învățare.

Pasul 3: Planificarea calendaristică orientativă (nivelul 1)

- Se întocmește la începutul semestrului/anului școlar în baza unităților de învățare și a elementelor corelate, identificate la pasul 2 .
- Formează un *cadru* care să permită adecvarea demersului didactic la situația din clasă. (Primul nivel de proiectare; tabelul 4.)

Exemple: Informatica clasa XII. Proiectarea de lungă durată tradițională și pe unități de învățare:

Tabelul 3. Proiectarea tradițională la Informatică, clasa XII (o variantă admisibilă)

Nr.	Indicatorii competențelor specifice (CS) și subcompetențelor		Conținuturi	Nr. de ore	Data	Ob-ser-va-ți
	CS	SC				
Elemente de modelare (10 ore)						
1	CS1 CS4 CS5 CS6 CS9	• aplicarea criteriilor de clasificare a modelelor; • elaborarea modelelor matematice; • motivarea importanței modelării în activitatea economică și viața socială; • identificarea soluțiilor analitice și soluțiilor de simulare; • selectarea tipului soluției în funcție de natura problemei; • planificarea și realizarea procesului de rezolvare a unei probleme la calculator;	Clasificarea modelelor: materiale/ideale/matematice. Modelare matematică. Soluții analitice și soluții de simulare. Etapele rezolvării problemelor la calculator: formularea problemei/elaborarea modelului matema- tic/elaborarea algoritmului/scrierea programului/testa- rea programului/analiza și interpretarea rezultatelor. Evaluare	2 2 2 2 2		
Calcul Numeric (24 ore)						
2	CS1 CS4 CS5 CS6 CS7 CS9 CS10	• identificarea valorilor exacte și a aproximărilor acestora; • determinarea erorii absolute și a erorii relative; • evaluarea erorilor de calcul, generate de erorile datelor de intrare; • estimarea erorilor, generate de particularitățile reprezen- tării numerelor în calculator;	Erori în calculele numerice: eroarea absolută/eroarea relativă. Erori ale datelor de intrare/de rotunjire/de aproxima- re/*de metodă/*de problemă; Evaluare	1 1 1		
		• utilizarea algoritmilor elementari pentru separarea solu- țiilor pe un interval dat; • identificarea condițiilor de aplicare a metodei biseecției (coardelor, Newton); • elaborarea într-un limbaj de programare de nivel înalt a pro- gramelor de calcul iterativ al soluției ecuației algebrice sau transcendente prin metoda biseecției (coardelor, Newton); • alegerea metodei de rezolvare a ecuațiilor algebrice și transcendente (biseecției, coardelor, Newton) adecvate pen- tru o problemă dată;	soluția ecuației; separarea soluțiilor; metoda biseecției formule recurente/aproximare inițială; metoda coardelor/extremități fixe; metoda lui Newton. Evaluare	2 2 2 1		

		• selectarea tehnicii de implementare a algoritmului; • elaborarea subprogramelor pentru calculul numeric al determinanților; • elaborarea într-un limbaj de programare de nivel înalt, a subprogramelor pentru rezolvarea sistemelor de ecuații liniare;	Calculul determinanților algoritmi recursivi/algoritmi iterativi. Regula lui Cramer. * Metoda lui Gauss.	2		
			Evaluare	2		
		• elaborarea programelor (subprogramelor) pentru calculul numeric al integralelor prin metoda dreptunghiurilor în funcție de un număr de divizări, stabilit apriori; • identificarea problemelor a căror rezolvare se reduce la calculul unei integrale definite;	Calculul numeric al integralelor. Metoda dreptunghiurilor: de mijloc/de stînga/de dreapta. * Metoda trapezelor.	2		
			Evaluare	2		
				1		
3. Baze de date						
3	CS1 CS4 CS5 CS7 CS10 CS11	• descrierea structurii bazelor de date ierarhice, în rețea și relaționale; • descrierea structurii și a funcțiilor sistemelor de gestiune a bazelor de date; • distingerea etapelor de elaborare a unei baze de date; • cunoașterea rolului persoanelor antrenate în elaborarea și utilizarea bazelor de date; • proiectarea structurii tabelelor bazei de date; • crearea tabelelor cu ajutorul sistemului de asistență sau prin proiectare independentă; • utilizarea metodelor de introducere a datelor în tabele; • crearea și editarea tabelelor pentru problemele frecvent întâlnite în matematică, fizică, biologie, chimie, geografie etc.; • stabilirea corelațiilor între tabele; • utilizarea operațiilor destinate sortării înregistrărilor, căutării și înlocuirii valorilor; • elaborarea filtrelor pentru selectarea înregistrărilor; • alegerea tipurilor de interogări adecvate pentru prelucrarea datelor; • elaborarea interogărilor cu ajutorul sistemului de asistență sau prin proiectare independentă;	Noțiuni și concepte. Tipuri de baze de date: ierarhice/în rețea/relaționale. Sisteme de gestiune a bazelor de date: structura/funcțiile. Etapale de elaborare a unei baze de date.	1		
				1		
				1		
				1		
				1		
				1		
			Tabele: destinația și structura; crearea/introducerea datelor; modificarea structurii/editarea înregistrărilor; stabilirea corelațiilor între tabele; sortarea înregistrărilor/căutarea și înlocuirea valorilor/ crearea și folosirea unui filtru.	1		
				1		
				1		
				1		
				1		
			Evaluare Interogări: destinația și structura interogărilor; crearea unei interogări/sortarea și gruparea înregistrărilor.	1		
				1		

	• elaborarea interogărilor pentru selectarea datelor; • identificarea componentelor unei expresii și aplicarea regulilor de formare a expresiilor; • elaborarea interogărilor de acțiune; • utilizarea tehnicilor de grupare și totalizare a datelor; • identificarea elementelor care alcătuiesc formularele; • elaborarea formularelor cu ajutorul sistemului de asistență; • aplicarea tehnicilor de modificare a formularelor; • utilizarea formularelor pentru vizualizarea, modificarea și validarea datelor; • elaborarea formularelor pe baza tabelelor corelate; • identificarea componentelor care alcătuiesc raportul; • elaborarea rapoartelor și subrapoartelor cu ajutorul sistemului de asistență sau prin proiectare independentă; • utilizarea tehnicilor de modificare a rapoartelor; • utilizarea tehnicilor de grupare a datelor într-un raport;	Expresii: construirea expresiilor; interogări pentru adăugare, actualizare și eliminare; gruparea și totalizarea datelor în interogare. Evaluare Formulare: destinația și structura/crearea și funcționarea unui formular/modificarea formularelor/folosirea formularelor; crearea formularelor pe baza tabelelor corelate/formatarea datelor. Rapoarte: destinația și structura rapoartelor. Crearea și funcționarea unui raport/crearea rapoartelor pe baza tabelelor corelate. gruparea și totalizarea datelor într-un raport. Evaluare * Compactarea și repararea bazei de date. * Securitatea unei baze de date/* Administrarea unei baze de date.	2		
Elemente de web design					
CS1 CS7 CS8 CS10 CS11	• identificarea cerințelor și recomandărilor către documente Web; • cunoașterea etapelor de elaborare a unui document Web; • identificarea documentelor Web în format HTML; • elaborarea documentelor HTML cu ajutorul aplicațiilor de oficiu; • publicarea documentelor Web în Internet; • elaborarea documentelor HTML simple; • formatarea textului cu ajutorul instrumentelor HTML; • crearea și organizarea listelor cu ajutorul instrumentelor HTML; • crearea și utilizarea legăturilor interne și externe; • inserarea în documente HTML a imaginilor; • crearea și editarea tabelelor în documente HTML; • utilizarea tabelelor pentru amplasarea în pagină a elementelor HTML;	Documente Web: noțiuni și concepte/formatele documentelor Web/structura unui site/cerințe către documente Web/etapele de elaborare a documentelor Web/Resurse soft pentru crearea și generarea documentelor Web/crearea documentelor HTML cu ajutorul aplicațiilor de oficiu/publicarea documentelor Web în Internet. Evaluare Structura generală a unui document HTML. Formatarea textului titlu/paragrafe/comentarii/stiluri fizice/stiluri logice/linii. Liste: ordonate/neordonate/de definiții. Tipuri de marcare/Tipuri de numerotare. Liste imbricate. Evaluare Legături externe, interne: referință/cale/ancoră/comentariu la legătură. Imagini: dimensiuni; chenar; legături; comentarii. Tabele: titlu/linie/coloană/celulă/contur. Evaluare	1		
			1		
			1		
			2		
			2		
			1		
			2		
			1		
			2		
			1		
			2		
			1		

Proiectarea de lungă durată (pe unități de învățare):
Tabelul 4. Proiectarea de lungă durată pe unități de învățare

Nr	CS	Unitatea de învățare	Conținuturi vizate	Nr. ore	Data	Observații
1	CS1 CS4 CS5 CS6	Elemente de modelare	Clasificarea modelelor. Modelare matematică. Soluții analitice și soluții de simulare. Etaple rezolvării problemelor la calculator.	10		
2	CS7 CS10	Erori în calcul numeric	Erori în calcul numeric	3		
3	CS11	Rezolvarea ecuațiilor algebrice și transcendente	Soluția ecuației. Separarea soluțiilor. Metoda biseției. Metoda coardelor. Metoda lui Newton.	9		
4		Determinanți și sisteme de ecuații liniare	Calculul determinanților. Regula lui Cramer. * Metoda lui Gauss.	7		
5		Calculul integralei definite	Calculul numeric al integralelor. Metoda dreptunghiurilor: de mijloc/de stînga/de dreapta. * Metoda trapezelor.	5		
6	CS1 CS7 CS8 CS10 CS11	Elaborarea structurii și tabelor bazelor de date	Noțiuni și concepte. Tipuri de baze de date. Sisteme de gestiune a bazelor de date. Etaple de elaborare a unei baze de date. Tabele.	8		
7		Interogări, expresii	Interogări, expresii	5		
8		Formulare, rapoarte	Formulare, rapoarte	7		
9		Documente Web. Concepte de bază	Documente Web. Crearea documentelor Web cu ajutorul aplicațiilor de oficiu.	4		
10		Prelucrarea textului în documentele HTML	Structura generală a unui document HTML. Prelucrarea textului în limbajul HTML.	5		
11		Elemente externe în documentele HTML	Legături, imagini, tabele în documentele HTML.	5		

Proiectarea didactică a lecției (unității de învățare) Nivelul 2 de proiectare

Proiectarea didactică a lecției

În contextul celor menționate anterior, pentru modelul clasic de proiectare în bază de lecții pot fi utilizate în continuare modelele de structură existente ale proiectelor didactice.

Elementul de transformare în acest caz apare prin indicarea în antetul proiectului didactic a subcompetențelor și competențelor specifice, corelate cu conținuturile didactice asociate lecției.

Proiectarea didactică a unității de învățare

La fel ca în cazul proiectării tradiționale de lungă durată, vom cerceta mai detaliat proiectarea unei unități de învățare.

Se propune un cadru simplificat de proiectare, bazat pe următoarele etape:

- *Familiarizare* (Evocare, Explorare)
- *Structurare* (Explicare, Esențializare)
- *Aplicare* (Exersare, Extindere)

Fiecare etapă presupune realizarea unor activități comune ale profesorului și elevilor. Proiectul didactic va fi elaborat reiesind din acțiunile profesorului în cadrul activității concrete din cadrul unității de învățare/lecției.

Antetul proiectului didactic va conține informații cu referire la: unitatea de învățare (denumire), competențele transversale și specifice vizate, profesor, clasă, locul (locurile) desfășurării activităților, strategiile didactice (principii, metode, procedee), forme de organizare a activităților și alte informații semnificative din punctul de vedere al profesorului. Majoritatea informațiilor se regăsesc detaliate în tabelul de proiectare și pot fi doar nominalizate.

Tabelul activităților didactice va conține informații la datele realizării fiecărei lecții din cadrul unității de învățare, subcompetențele vizate, conținuturile didactice corelate cu acestea, activitățile didactice, care urmează să fie realizate pentru a procesa conținuturile, resursele materiale (de toate tipurile), metodologice (metode aplicate în cadrul activității didactice) și de timp (durata estimativă a activității), evaluări ale calității realizării activității și observații pentru îmbunătățirea ulterioară a structurii proiectului didactic.

10. Proiecte de lecții, centrate pe formarea la elevi a competențelor specifice disciplinei

Proiecte didactice de lecții independente

PROIECT DIDACTIC

Lecție de formare și consolidare a cunoștințelor

Disciplina: Informatica

Clasa: a XI-a

Profesor: Birsan Valentin

Unitatea de învățare: Tehnici de programare

Tema: Metoda Greedy

Tipul lecției: Formarea competenței de rezolvare a problemelor

Durata: 90 min

Locul de desfășurare: Laboratorul de informatică

Nivelul inițial al clasei (cunoștințe și deprinderi preliminare):

- elevii realizează algoritmi de sortare;
- elevii creează și utilizează subprograme;
- elevii efectuează citirea datelor din fișiere text.

Competențe specifice:

- Elaborarea modelelor informatice ale obiectelor, sistemelor și proceselor frecvent întâlnite în activitatea cotidiană.
- Aplicarea metodelor de algoritimizare, de formalizare, de analiză, de sinteză și de programare pentru soluționarea problemelor legate de prelucrarea automatizată a informației.
- Traducerea algoritmilor frecvent utilizați într-un limbaj de programare de nivel înalt.

Subcompetențe:

- Estimarea complexității algoritmilor.
- Utilizarea tehnicilor de programare la rezolvarea problemelor din diferite domenii.

Resurse materiale:

- Sala de calculatoare cu rețea locală/acces Internet;
- Proiector multimedia/sisteme de reproducere a sunetului.

Resurse didactice digitale:

- Program de control și transfer de date în rețeaua locală NetOpSchool;
- Prezentări necesare în diferite momente ale lecției;
- Secvența de film „Fata babei și fata moșneagului” (fragmentul alegerii cușăului de fata babei) www.telecinemateca.com/fata-babei-si-fata-mosneagului;
- Prezentarea „Metoda Greedy” <http://www.slideshare.net/sorlat/metoda-greedy>;

- Compilator FreePascal. Subprograme pentru: sortarea unui tablou liniar; citirea din fișier text a valorilor elementelor unui tablou liniar.

Metode de evaluare:

- Întrebări orale, testare la calculator.

Scenariul desfășurat al lecției

Momentul	Du- rata	Etapă	Activatate/Descrierea activității	Resurse	Comen- tarii
1	2		3	4	5
Moment or- ganizatoric	2		– Atenționarea elevilor; – Notarea elevilor absenți; – Pregătirea celor necesare pentru lecție.		Crearea atmosferei de lucru în clasă.
Momentul 1. Introducerea în temă	5-8	Evo- care	Profesorul propune elevilor să urmărească o secvență a filmului „Fata babei și fata moșneagului”. Procedura de alegere a cufărului de către fete. (30 s – 1 min) Se lansează discuția: fiecare fată a făcut alegerea după anumite principii – a selectat <i>cel mai potrivit</i> obiect după un careva criteriu. Dacă condiția impusă permitea alegerea a K cufere, pe care le alegea fiecare dintre fete? (primele K mai mari/mai mici) Cum pot fi selectate eficient cele mai potrivite K? (prin sortare) Cuferele au culori diferite. Dacă fetele nu acceptă cuferele de o anumită culoare, cum se va face alegerea? (se aleg primele K cufere, care nu au culoarea „interzisă”) Se stabilește motivul denumirii metodei „Greedy” (trad. „La-com”).	Calculator, proiector, boxe. Secvență film.  Poster/electronic sau pe hîrtie pentru notarea punctelor-cheie ale discuției.	Introdu- cerea în metodă prin ana- liza unei situații cunos- cute.

Momentul 2. Generaliza- rea metodei	10	Ex- plo- rare/ Struc- turare	Elevii, împreună cu profesorul, formulează metoda: Fie că există o mulțime din N elemente, unele dintre care vor forma soluția unei probleme. Soluția se formează în modul următor: din mulțimea de elemente se alege acel care corespunde cel mai mult unui criteriu de includere. El este inclus în soluție și exclus din mulțimea cercetată. Procesul de alegere a celui mai potrivit element și adăugare a lui în soluție se repetă atît timp cît este permis de condițiile problemei. Dacă elementul curent ales pentru includere contrazice anumite restricții impuse de problemă, el este exclus din mulțimea cercetată fără a fi inclus în soluție. Abstractizarea: expunerea metodei în termeni matematici: prezentarea „Metoda Greedy”.	Poster/electronic sau pe hirtie pentru descrierea metodei. Prezentarea: 	
Momentul 3 Funcționa- lizarea/inte- riorizarea	20	Exer- sare	Problemă prototip <i>Plata unei sume cu bancnote de valori date (număr infinit de bancnote)</i> Se identifică posibilitatea de aplicare a metodei Greedy. Se stabilesc etapele de rezolvare a problemei: 1. Citirea datelor inițiale (valorile bancnotelor, suma pentru plată); 2. Sortarea valorilor bancnotelor în ordine descrescătoare; 3. Selectarea nominalului maxim; 4. Folosirea bancnotelor de nominal curent atît timp cît suma pentru plată rămîne mai mare sau egală cu nominalul; 5. Dacă suma curentă este diferită de 0, se trece la bancnotele de nominal următor și se repetă pasul 4; în caz contrar Stop, suma a fost plătită. Dacă toate tipurile de nominal au fost folosite, dar suma nu a devenit 0, STOP – soluția nu există. Rezolvarea e realizată în paralel, folosind subprogramele realizate anterior.	Ciobanu I., Corlat S., Malearovici A. ș.a., <i>Bacalaureat 2003. Informatică</i>, Chișinău, Lyceum, 2003, prob. 1, p. 63. Calculatoare individual/perechi Compilatorul FreePascal. Subprogramele elaborate anterior.	

Momentul 4 Analiza complexității metodei	10	Extindere	Se analizează complexitatea fiecărei etape a algoritmului: Citirea datelor/liniar față de numărul elementelor introduse. Sortarea elementelor/ în funcție de algoritmul de sortare utilizat – $O(N^2)$ sau $O(N \log N)$. Verificarea condiției, dacă elementul selectat poate fi inclus în soluție. Poate varia de la problemă la problemă, dar de obicei este liniară față de N . Pentru N elemente complexitatea sumară va fi – $O(N^2)$. *Prin urmare, în presupunerea unei verificări liniare a condiției de includere, complexitatea algoritmului va fi de – $O(N^2)$. În caz general, cînd verificarea are complexitatea – $O(N^k)$, complexitatea generală a algoritmului va fi de $O(N^{k+1})$.	Programul elaborat anterior. Manualul de informatică pentru clasa a XI-a. Compartimentul „Analiza complexității algoritmilor”. Calculatoare individual/perechi.	Generarea se deduce în condiția prezenței rezervei de timp.
Momentul 5 Exteriorizare	25	Exersare	Rezolvarea problemei rucsacului: 1. Stabilirea criteriului de optimalitate: raportul preț/volum pentru fiecare obiect; 2. Sortarea obiectelor în descrescere după raportul preț/volum; 3. Adăugarea consecutivă a obiectelor sortate în rucsac, dacă a) volumul liber în rucsac este mai mare sau egal decît volumul obiectului curent; b) mai există obiecte, care încă nu au fost „cercetate”; 4. Calculul sumei costurilor obiectelor incluse. Evaluare: verificarea progresului/rezultatelor.	Sergiu Corlat, Lilia Ivanov, <i>Calcul numeric. Curs de lecții</i> , Chișinău, CCRE „Presa”, 2004, pag. 19. Calculatoare individual/perechi.	
Momentul 6 Analiza optimalității metodei	8	Extindere	Pentru setul de date indicat în sursa bibliografică, se va determina prin calcule manuale soluția cea mai bună. Se va compara cu cea propusă de metoda Greedy. Discuție : de ce apar diferențe ? Ce rezultă ?	Seturi de date de intrare, pix, caiet pentru calcule.	
Momentul 7	2	Extindere	Tema pentru acasă: <i>Plata unei sume cu bancnote de valori date (număr finit de bancnote).</i> Organizarea optimă a reprezentațiilor de teatru (problema segmentelor).	Ciobanu I., Corlat S., Malearovici A. ș.a., <i>Bacalaureat 2003. Informatică</i> , Chișinău, Lyceum, 2003, prob. 2, prob. 3, p. 63.	Textul temei pentru acasă poate fi amplasat pe un blog sau site pentru accesul liber al elevilor

Proiectarea didactică a unității de învățare

Disciplina: Informatica

Clasa: a XII-a

Profesor:

Unitatea de învățare: Prelucrarea textului în documentele HTML

Tipul lecțiilor: Lecțiile 1,3: achiziționare/funcționalizare a cunoștințelor (activități de explorare); Lecțiile 2,4: interiorizare/exteriorizare (activități practice) ;
Lecția 5: evaluare.

Locul de desfășurare: Laboratorul de informatică

Nivelul inițial al clasei (cunoștințe și deprinderi preliminare):

- elevii cunosc noțiunile de bază din domeniul designului Web;
- elevii creează documente Web simple folosind aplicații de oficiu;
- elevii folosesc editoare de texte simple și programe de navigare Web.

Competențe specifice:

CS8. Crearea și elaborarea documentelor Web.

Subcompetențe:

- elaborarea documentelor HTML simple;
- formatarea textului cu ajutorul instrumentelor HTML;
- crearea și organizarea listelor cu ajutorul instrumentelor HTML.

Resurse materiale:

- Sala de calculatoare cu rețea locală și acces Internet;
- Proiector multimedia, conectat la calculator.

Resurse didactice digitale:

- Aplicații de gestionare a transferului de date în rețeaua locală/globală;
- Teste locale/online elaborate apriori;
- Prezentări necesare la diferite activități ale lecțiilor din unitatea de învățare;
- Modele de documente Web, elaborate în scop instructiv.

Resurse bibliografice/Web:

- *Informatică. Curriculum pentru clasele a X-a – a XII-a*, Chișinău, Știința, 2010;
- Anatol Gremalschi, Sergiu Corlat, Andrei Braicov, *Informatică. Manual pentru clasa a XII-a*, Știința, Chișinău, 2010;
- Andrei Braicov, *HTML. Ghid de inițiere*, Chișinău, Editura Prut Internațional, 2008;
- www.w3c.org

Metode de evaluare:

- testare la calculator;
- întrebări orale;
- activități practice;

Scenariul desfășurat al lecțiilor în cadrul unității de învățare

Data	Detalii conținut	Activități de învățare	Resurse (materiale, procedurale, de timp)	Evaluare	Observații
Lecția 1		Activități organizatorice	2 min		
	Fixare conținuturi. Structura generală a unui document HTML: – secvență de control. Formatarea textului: – titlu; – paragrafe; – comentarii; – stiluri fizice; – stiluri logice. Linii orizontale.	Evocarea: prezentarea procesului de elaborare a paginilor Web cu resurse HTML. Explorarea: Discuția resurselor prezentate: Care este deosebirea între documentul HTML vizualizat într-un editor de texte și într-un program de explorare? Care este structura unui document HTML simplu? Care este rolul secvențelor de control? Cum pot fi identificate secvențele de control?	Secvență film 30” sau o pagină web simplă în web și codul ei. – 10 min total	Pot oare elevii să stabilească diferențele între pagina web și sursa ei? Pot oare elevii să explice pentru ce e necesar limbajul HTML? Pot oare elevii să identifice informația primară în documentele HTML? Pot oare elevii să identifice secvențele de control?	
	Funcționalizare cunoștințe.	Explorare/Structurare: stabilirea compartimentelor din care este format un document Web, identificarea secvențelor de control necesare pentru delimitarea structurii. Crearea de către elevi a unui document HTML vid. Adăugarea titlului în document. Vizualizarea și analiza rezultatelor.	Resurse: Diapozitive sau scheme cu descrierea structurii și conținutului documentelor HTML. Calculator – individual/perechi. Editor de texte (Notepad), Browser (Internet Explorer). Timp – 10 min	Pot oare elevii să identifice elementele de structură ale unui document HTML? Pot oare elevii să creeze un document HTML cu un titlu dat?	

		Explorare/Structurare/Exersare: stabilirea modurilor de separare a textului, evidențiere a fragmentelor de text și a paragrafelor. Completarea către elevi a conținuturilor documentului HTML creat anterior. <Hx> </Hx> Crearea unui subtitlu. <P> </P> Crearea unui paragraf. Alinierea în paragraf. Evidențierea textului în paragraf: , <I> </I>, <U> </U>, etc. Elemente de stiluri logice: , , <KBD> </KBD> Separatori (linii orizontale). <HR> și atributele ei. Vizualizarea și analiza rezultatelor.	Resurse: Diapozitive sau scheme cu descrierea structurii și conținutului documentelor HTML. Calculator – individual/perechi. Editor de texte (Notepad), Browser (Internet Explorer). Timp – 20 min	Pot oare elevii să introducă în documentele HTML secvențe de text și: – să le separe în paragrafe? – să formeze subtitluri? – să evidențieze textul? – să alinieze textul? – să aplice un stil la paragraf?	
		Extindere: (tema pentru acasă) crearea paginilor care vor forma proiectul selectat la unitatea de învățare anterioară. Căutarea informațiilor pentru plasare.	Resurse: Manualul de informatică, cl. a XII-a Acces rețea Internet, bibliotecă. Timp: 3 min (clasă) 1 – 2 ore (pregătire independentă).	Au oare elevii competența de a elabora structura unui site? Au oare elevii competența de a selecta informații la diferite teme din surse bibliografice și Internet?	
Lecția 2		Activități organizatorice	2 min		
	Interiorizare	Structurare/Exersare/Extindere: elaborarea documentelor HTML și a conținutului text al acestora pentru proiectul individual/de grup. Formatarea textului introdus. Extinderea posibilităților de formatare cu secvența și atributele ei.	Resurse: Calculator – individual/perechi, Editor de texte (Notepad), Browser (Internet Explorer). Timp – 30 min	Pot oare elevii aplica cunoștințele acumulate pentru a crea documente HTML și a formata textul din ele?	
	Exteriorizare				

		Evaluare: aprecierea calității formătărilor textului documentelor HTML create. (evaluarea de către profesor, autoevaluarea, evaluarea reciprocă)	Timp – 10 min	Pot oare elevii aprecia obiectiv calitatea activităților realizate de ei sau de colegii lor? În ce măsură pot utiliza elevii elementele de formatare a textului?	
		Extindere: (tema pentru acasă) completarea paginilor proiectului individual/de grup cu informații text. Identificarea în paginile Web a altor metode de formatare a textului.	Resurse: Manualul de informatică, cl. XII. Acces rețea Internet, bibliotecă. Timp: 3 min (clasă) 1 – 2 ore (pregătire independentă).		
Lec- ția 3	Liste: ordonate; neordonate; de definiții. Tipuri de marcare. Tipuri de numerotare.	Activități organizatorice	2 min		
		Evocarea: prezentarea procesului de creare a listelor cu resurse HTML/identificarea listelor și a tipologiei lor în exemple. Explorarea: Discuția resurselor prezentate: Care sînt metodele noi de formatare a textului, metode identificate? Ce tipuri de liste pot fi identificate? Cum se marchează o listă cu marcare și elementele ei? Cum se marchează o listă cu numerotare și elementele ei? Cum se marchează o listă de definiții și elementele ei?	Secvență video 1 – 2’ sau prezentarea unei pagini Web cu liste. Pagini Web cu liste și codurile sursă ale paginilor/diapositive, postere etc. – 10 min total	Pot oare elevii identifica listele în documentele HTML? Pot oare identifica tipul de listă? Elementele listei?	

		Exersare/Structurare/Extindere: stabilirea modurilor de organizare a listelor. Completarea către elevi a conținuturilor documentului HTML pentru exersare, creat anterior. Crearea listelor cu marcate. Atributul TYPE. Crearea listelor cu numerotare. Atributele TYPE, START. <DL> </DL> <DT> <DD> Crearea listelor de definiții. Vizualizarea și analiza rezultatelor	Resurse: Diapozitive sau scheme cu descrierea structurii listelor pentru documentele HTML. Calculator – individual/perechi. Editor de texte (Notepad), Browser (Internet Explorer). Timp – 20 min	Pot oare elevii să creeze liste/cu marcate/numerotare/de definiții? Pot să seteze proprietățile listei create?	
		Extindere: (tema pentru acasă) colectarea textului structurat în liste pentru paginile proiectului individual/de grup.	Resurse: Manualul de informatică, cl. XII. Acces rețea Internet, bibliotecă. Timp: 3 min (clasă) 1 – 2 ore (pregătire independentă).		
Lecția 4	Liste imbricate.	Activități organizatorice	2 min		
	Fixare cunoștințe				
	Funcționalizare	Evocare: prezentarea unui document cu liste imbricate. Discuția posibilităților de realizare a listelor imbricate.	Resurse: Diapozitive sau scheme cu liste imbricate în documentele HTML. 5 min	Pot oare să genereze cunoștințe noi? Folosirea listelor imbricate.	
	Interiorizare				
	Exteriorizare	Structurare/Exersare/Extindere: Structurarea conținutului text al documentelor HTML prin adăugarea listelor/cu marcate/cu numerotare/de definiții. (proiectul individual/de grup). Experiment – imbricarea listelor.	Resurse: Calculator – individual/perechi, Editor de texte (Notepad), Browser (Internet Explorer). Timp – 25 min	Pot oare elevii să aplice cunoștințele obținute anterior pentru a crea propriile documente Web?	

		Evaluare: aprecierea calității formătărilor textului documentelor HTML create. (evaluarea de către profesor, autoevaluarea, evaluarea reciprocă)	Timp – 10 min	Pot oare elevii aprecia obiectiv calitatea activităților realizate de ei sau de colegii lor? În ce măsură pot utiliza elevii listele ca element de formatare a textului?	
		Extindere: (tema pentru acasă) colectarea textului structurat în liste imbricate și amplasarea în paginile proiectului individual/de grup.	Resurse: Manualul de informatică, cl. a XII-a. Acces rețea Internet, bibliotecă. Timp: 3 min (clasă) 1 – 2 ore (pregătire independentă).		
Lec- ția 5	Evaluare	Activități organizatorice	2 min		
		Evaluare: (test + proiect) 1. Itemi, exerciții și probleme pentru verificarea aptitudinilor de creare a documentelor HTML și de structurare a textului în ele; 2. Proiectul individual la etapa curentă de realizare, transmis pe adresa de mail indicată de profesor.	Resurse: Calculator – individual; Browser (Internet Explorer); Acces Internet; Aplicații EAC online/local AeL, Lemill, Moodle sau teste tipărite. Timp – 40 min	În ce măsură au realizat elevii scopurile unității de învățare – formatarea textului? Care sînt itemii calitativi ai testului și care trebuie înlocuiți?	
		Extindere: (tema pentru acasă) identificarea elementelor externe în paginile Web.	Resurse: Acces rețea Internet, bibliotecă. Timp: 3 min (clasă) 1 – 2 ore (pregătire independentă)		

Note:

1. Structura, tipul și numărul activităților poate varia în funcție de clasă, profesor, dotare tehnică.
2. Rubrica (coloana) evaluare se referă la autoevaluare de către profesor a calității activităților realizate, nu la evaluarea elevilor.

11. Anexe: modele de realizare a instruirii în bază de problemă, în bază de proiect, studii de caz

Anexa 1. Instruirea în bază de problemă

Exemplu. Prelucrarea tablourilor bidimensionale.

Problemă:

Pentru a determina zonele cu pericol sporit de inundare, agenția Apele Moldovei a efectuat cartografierea unor regiuni dreptunghiulare ale luncii Prutului. Fiecare regiune cartografiată a fost divizată în zone pătrate cu latura de 500 m (figura 1). O zonă a regiunii este considerată „de risc”, dacă prin ea trece albia râului. În caz contrar zona este considerată „de siguranță”. Agenția vrea să stabilească suprafața totală a zonelor de risc în fiecare regiune.

Deoarece volumul imaginilor este foarte mare, specialiștii agenției au marcat în fiecare regiune zonele de risc cu 1, iar celelalte cu 0 (figura 2) și au păstrat doar tabelele astfel create (figura 3).

Pentru o regiune cartografiată numărul de zone atât pe orizontală, cât și pe verticală nu depășește 20.

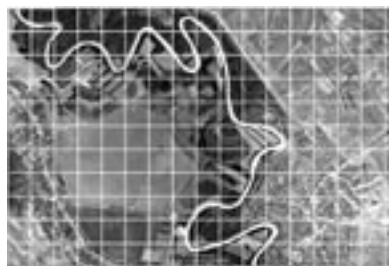


Fig. 1

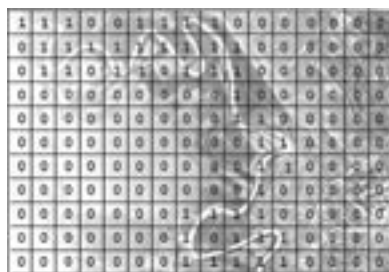


Fig. 2

Discuții:

1. Cum se rezolvă problema? (R: Trebuie de aflat numărul zonelor de risc, apoi de înmulțit cu suprafața unei zone.)
2. Se poate rezolva manual? (R: Da!)
3. Dar dacă avem mai multe regiuni? Dar dacă numărul de zone e mai mare? (R: Scriem un program!)
4. Ce ne trebuie pentru aceasta, cum transmitem tabelul-hartă? (R: Folosim un tablou bidimensional!)
5. Cum introducem datele?/pentru exemplul din imagine volumul de date care urmează să fie introduse este destul de mare, deci ar fi logic ca problema să se discute după studierea fișierelor text. (R: De la tastatură/din fișier.)

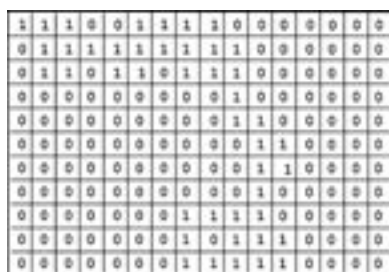


Fig. 3

6. Cum numărăm zonele de risc? (R: Parcurgem toate liniile de la prima coloană până la ultima și la întâlnirea unui element cu valoarea 1 incrementăm numărul zonelor de risc. Inițial numărul zonelor de risc se va considera 0.)
7. Cum calculăm suprafața totală a zonelor de risc? (R: Fie K – numărul zonelor de risc în regiune, L – lungimea laturii unei zone în km. Atunci suprafața totală a zonelor de risc va fi $K \cdot L \cdot L \text{ km}^2$.)

Algoritm:

Fie regiunea cercetată are M zone în lungime și N zone în înălțime. Latura unei zone are lungimea L . Datele se introduc în tabloul bidimensional A cu N linii și M coloane.

Precondiții: Se introduc valorile N , M , L , valorile elementelor tabloului.

- Pas 1. $K \leftarrow 0$
- Pas 2. Fie I – indicele liniei curente. $I \leftarrow 1$.
- Pas 3. Pentru toți J de la 1 la M repetăm: dacă $A[I, J]$ este egal cu 1, atunci $K \leftarrow K + 1$
- Pas 4. Dacă $I < N$ (nu am parcurs încă toate liniile), atunci $I \leftarrow I + 1$ și revenim la Pas 3, altfel trecem la Pas 5
- Pas 5. $S \leftarrow K \cdot L \cdot L$
- Pas 6. Afișăm S . SFÎRȘIT.

Realizarea problemei într-un limbaj de programare de nivel înalt nu prezintă dificultăți și e lăsată în seama cititorului.

Analiza activităților pentru rezolvarea problemei:

1. Elevii discută o situație (problemă) reală.
2. Elevii observă procesul de abstractizare a problemei, legătura între obiectul real, continuu și structura discretă, care îl reprezintă.
3. Elevii stabilesc tipul de date necesar pentru a transmite structura discretă în viitorul program și proprietățile acestui tip.
4. Elevii construiesc algoritmul de rezolvare.
5. Elevii scriu programul, care realizează algoritmul într-un limbaj de programare de nivel înalt.
6. Elevii verifică rezultatele obținute, le discută, propun situații reale care pot fi reduse la aceeași problemă.
7. Profesorul consultă elevii la fiecare dintre etapele menționate, iar în caz de necesitate – intervine cu indicații sau soluții de etapă.

Subcompetențe vizate:

- Argumentarea necesității structurării datelor.
- Prelucrarea datelor structurate.
- Alegerea structurii de date, adecvate rezolvării unei probleme.

Competențe specifice dezvoltate:

- CS1. Formarea unei viziuni științifice asupra componentei informatice în societatea contemporană.
- CS2. Cunoașterea proceselor, principiilor și metodelor de codificare și decodificare a informației în scopul realizării comunicării interumane și umane – sistem informatic.
- CS3. Elaborarea modelelor informatice a obiectelor, sistemelor și proceselor frecvent întâlnite în activitatea cotidiană.
- CS4. Aplicarea metodelor de algoritmizare, de formalizare, de analiză, de sinteză și de programare pentru soluționarea problemelor legate de prelucrarea automatizată a informației.
- CS5. Translarea algoritmilor frecvent utilizați într-un limbaj de programare de nivel înalt.

Anexa 1b**Proiect: Compararea eficienței algoritmilor de sortare**

Proiectul poate fi propus cu diferite nivele de complexitate, în funcție de algoritmii de sortare selectați pentru analiza comparativă.

Pentru un proiect de complexitate mică pot fi propuși algoritmi de sortare prin metoda bulelor și algoritmul de sortare prin selecție (sau inserție). Proiectul de complexitate medie presupune compararea unui algoritm lent (metoda bulelor, sortarea prin selecție, sortarea prin inserție) și a unui algoritm de sortare optim (sortarea rapidă [*quicksort*], sau heap sort). Complexitatea sporită a proiectului se obține la compararea eficienței a doi algoritmi optimali – sortarea rapidă și heap sort (sau sortarea rapidă și merge sort).

Pentru realizarea proiectului elevul (sau grupul de elevi) va avea nevoie de resurse:

- A. Informaționale – descrierile algoritmilor pentru metodele selectate; descrierile algoritmului de măsurare a timpului de execuție a algoritmului; algoritmul de alocare aleatorie a valorilor elementelor unui tablou;
- B. Hardware/comunicaționale: calculator, acces la rețeaua Internet, proiector multimedia;
- C. Software: compilator al unui limbaj de programare de nivel înalt Free-Pascal, Dev C++ sau echivalente.

Durata estimativă a proiectului: 7 – 14 zile.

Perioada de realizare:

- după studierea unității de învățare *Analiza algoritmilor* (nivelul elementar);
- după studierea unității de învățare *Metoda desparte și stăpânește* (nivelul mediu);
- nu se fixează (nivelul avansat).

Etapale proiectului:

Etapa 0 (realizată de profesor): selectarea temei, resurselor, formarea setului de indicații pentru echipa de proiect, a orarului prealabil.

Etapa 1 (ziua 1): formarea echipei, stabilirea scopului proiectului, repartizarea responsabilităților pentru realizarea proiectului, explicarea noțiunilor, cerințelor, metodelor de elaborare a proiectului.

Etapa 2 (ziua a 8-a): răspunsuri la întrebări, sugestii, explicații, verificarea progresului curent.

Etapa 3 (ziua a 15-a): prezentarea de către echipă a proiectului, evaluarea, discuția în cadrul clasei.

Exemplu: Compararea eficienței algoritmilor de sortare: metoda bulelor versus quicksort

Notă: Pentru prezentarea publică a proiectului, elevii vor folosi un proiector multimedia și un sistem pentru prezentări electronice (Power Point [MS Office] , Impress [Open Office] sau altele). Pentru a folosi eficient spațiul, nu vom include în descrierea curentă diapozitivele care însoțesc prezentarea, dar numai etapele prezentării, referințele la resursele utilizate și unele dintre materiale.

- Descrierea scopului proiectului:** Analiza eficienței algoritmilor de sortare în studiu pentru dimensiuni mari ale datelor sortate. Elaborarea recomandărilor pentru utilizarea fiecărui algoritm.
- Descrierea surselor de documentare și a resurselor de programare:** Pentru realizarea proiectului au fost studiați algoritmi de sortare prin metoda bulelor (<http://en.wikipedia.org/wiki/Bubblesort>) și algoritmul de sortare rapidă (<http://en.wikipedia.org/wiki/Quicksort>). Compilatorul Free Pascal a fost selectat din două motive: este un *compilator distribuit gratuit* și *permite crearea tablourilor liniare de dimensiuni mari* prin operații standard.
- Descrierea metodei de cercetare:** se compară repetat timpul de lucru al fiecăruia dintre algoritmi de sortare propuși pe seturi identice de date. Pentru aceasta fiecare algoritm se programează separat, în cadrul unei proceduri, se generează aleatoriu un tablou cu număr prestabilit de elemente și se transmite fiecărei proceduri de sortare. Timpul de lucru al algoritmului se stabilește ca modulul diferenței dintre timpul

curent la momentul apelului procedurii de sortare și a timpului curent la momentul revenirii în programul principal.

- d) *Prezentarea programului*: poate fi efectuată fragmentar, pe subprograme, cu indicarea doar a rolului fiecăruia dintre ele. Dacă proiectul este prezentat în cadrul unei activități extracurriculare (ore opționale, cerc, etc.) se poate alocă timp pentru discuția mai detaliată a acestuia.

Exemplu de realizare:

```

program p001;
uses crt, dos;
const n=30000;
type t=array[1..n] of integer;
var a,b: t;
    t1,t2: real;
    i:integer;

procedure bubble; {sortarea prin metoda bulelor}
var i,j,tmp: integer;
begin for i:=1 to n-1 do
  for j:=1 to n-i do
    if a[j]>a[j+1] then begin
      tmp:=a[j]; a[j]:=a[j+1]; a[j+1]:=tmp;
    end;
  end;

procedure qsort(st,dr:integer); {sortarea rapida}
var s,i,k,tmp: integer;
begin
  if (st < dr) then begin
    s:=b[st]; k:=st;
    for i:=st+1 to dr do
      if b[i]<s then begin k:=k+1; tmp:=b[i];
        b[i]:=b[k]; b[k]:=tmp;
      end;
    b[st]:=b[k]; b[k]:=s;
    qsort(st,k-1); qsort(k+1,dr);
  end
  else exit;
end;

function tc: real;
var h,m,s,ms: word;
begin
  gettime(h,m,s,ms);
  tc:=h*3600+m*60+s+ms/100;
end;

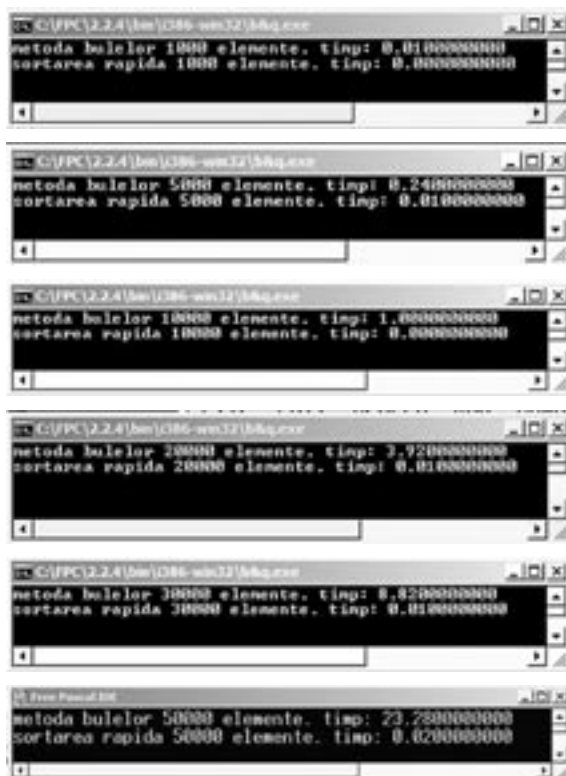
```

```

begin
clrscr;
{generarea aleatorie a tablourilor cu date pentru sortare}
randomize;
for i:=1 to n do
begin
a[i]:= 1+random(30000);
b[i]:=a[i];
end;
t1:=tc; {fixarea timpului inainte de apelul procedurii de sortare}
bubble;
t2:=tc; {fixarea timpului dupa apel}
writeln('metoda bulelor ', n, ' elemente. timp: ', t2-t1:0:10);
t1:=tc;
qsort(1,n);
t2:=tc;
writeln('sortarea rapida ', n, ' elemente. timp: ', t2-t1:0:10);
end.

```

e) Prezentarea rezultatelor:



Analiză: Se observă ușor creșterea continuă a timpului necesar pentru metoda bulelor, odată cu creșterea numărului de elemente sortate. Pentru sortarea rapidă timpul necesar aproape că nu crește, dar se observă că în unele cazuri timpul necesar pentru sortarea unor seturi mici de date poate să depășească timpul pentru sortarea seturilor mai mari.

Concluzii:

1. Algoritmul de sortare rapidă este în general mai eficient decât algoritmul de sortare prin metoda bulelor.
2. Pentru volume de date ce nu depășesc 1000 de elemente, alegerea algoritmului nu este esențială, pentru seturi mai mari și în restricții de timp impuse este util de folosit algoritmul de sortare rapidă.
3. Timpul necesar algoritmului de sortare rapidă nu este neapărat proporțional cu numărul de elemente. Eficiența algoritmului depinde și de amplasarea inițială a elementelor.
4. Pentru problemele reale, în care apare necesitatea sortării, este necesar de implementat algoritmi eficienți, chiar dacă aceasta cere un efort inițial mai mare. Exemple: sortarea datelor în registrele naționale ale mijloacelor de transport, ale populației etc.

Note:

1. Rezultatele obținute la realizarea de către elevi a algoritmilor pot să difere de cele prezentate din motivul generării aleatorii a valorilor și a parametrilor procesorului, la care se execută programul.
2. Modificarea numărului de elemente sortate se efectuează prin modificarea valorii constantei n în program.
3. Proiectul se transformă ușor într-un studiu de caz, dacă elevilor li se propun în calitate de resursă subprogramele care realizează algoritmi de sortare.

Anexa 1c: Studiu de caz „Publicarea documentelor Web”

Clasa a XII-a, profil umanist, profil real.

Studiul de caz poate fi propus atât în forma generală, cât și în variante partializate: *publicarea gratuită a documentelor Web*, *publicarea comercială a documentelor Web*, *publicarea documentelor Web cu ajutorul platformelor Web 2.0*. Selectarea variantei marchează și prima etapă a realizării studiului (etapa inițială: alegerea cazului).

Următoarea etapă (lansarea cazului) se inițiază în procesul activităților didactice, după realizarea de către elevi a unui document Web (fie cu ajutorul aplicațiilor de oficiu, fie cu instrumentele HTML), folosind conversația sau altă metodă didactică, în funcție de situația concretă.

Poate fi selectată o echipă sau câteva, pentru fiecare variantă separată a studiului.

Pentru realizarea studiului de caz elevii vor avea nevoie de resurse:

Resurse:

1. Calculator cu acces la rețeaua Internet;
2. Proiector multimedia (pentru prezentarea finală);
3. Sisteme de căutare a informației;
4. Enciclopedia electronică www.wikipedia.org;
5. Repozitoriul digital www.scribd.com (căutarea după cuvintele-cheie);
6. Resurse informaționale de pe site-ul www.molddata.md (compartimentele domeniului, hosting, legislație);
7. Resurse informaționale de pe site-ul www.moldtelecom.md (compartimentul Internet, servicii opționale);
8. Aplicații pentru crearea prezentărilor digitale;
9. Ned Snell, *Crearea paginilor Web*, Teora, 1999.

Scopuri:

1. Determinarea metodelor de publicare a documentelor Web.
2. Determinarea etapelor de publicare.
3. Stabilirea punctelor tari și a punctelor slabe pentru metodele de publicare identificate.
4. Prezentarea rezultatelor către auditoriu (clasă).

Perioada de realizare: compartimentul „elemente de Web design”

Durata: 1 săptămână

Ziua 1: Repartizarea și explicarea sarcinilor, identificarea resurselor informaționale, stabilirea modului de comunicare pentru perioada realizării studiului

Zilele 2 – 7. Consultări online sau directe cu autorii studiului (membrii echipei realizează etapele de procurare a informației, sistematizare a materialului).

Ziua a 8-a. Prezentarea de către autori a studiului pentru auditoriu (clasă). (etapele de prezentare, dezbateri, concluzii)

Exemplu de studiu prezentat:

Preliminarii

Fie că avem un site, pe care l-am proiectat, l-am programat și am verificat funcționalitatea. Teoretic, avem un site finalizat, dar care poate fi utilizat doar local. Cum se poate rezolva problema accesului la document al auditoriului căruia îi este adresat? Urmează să publicăm documentul pe Web, transformându-l într-un site funcțional.

Pentru aceasta trebuie să realizăm următoarele etape:

Atribuirea unui nume de domeniu;

Selectarea serverului Web pe care va fi stocat (găzduit) site-ul;

Organizarea și denumirea fișierelor, care vor forma site-ul în conformitate cu cerințele serverului gazdă;

Transferul fișierelor.

Numele de domeniu

Numele de domeniu este necesar pentru ca site-ul să aibă o identitate pe Web. Numele de domeniu va face parte din adresa URL a fiecărei pagini și va oferi site-ului o prezență distinctă pe Web.

Domeniile principale de pe Web pot fi de mai multe tipuri: comerciale **.com**; educaționale **.edu**; guvernamentale **.gov**; furnizorii de servicii de rețea **.net**; instituții non-profit **.org**; geografice **.md .ro** etc.

Numele de domeniu pentru site poate fi obținut contra plată sau gratuit.

Nume de domenii contra plată

Dacă site-ul aparține unei persoane juridice sau unei persoane fizice, care își dorește o prezență stabilă și credibilă pe Web, domeniul plătit este opțiunea perfectă. Înregistrarea unui domeniu nu este costisitoare, dar asigură identitatea și credibilitatea site-ului și proprietarului.

De exemplu, pentru domeniile **.md** înregistrate prin întreprinderea de stat MoldData, prețul este de aproximativ 500 lei pentru un an calendaristic (preț variabil la momentul realizării studiului).

În general există mai multe organizații care oferă servicii de repartizare a numerelor de domenii contra plată, de exemplu: www.host.md, www.10-domains.com etc.

În cazul numelui de domeniu plătit, adresa URL a site-ului va avea forma [http://www.\[numele de domeniu selectat\].com](http://www.[numele de domeniu selectat].com)

Domenii gratuite

Obținerea unui nume de domeniu gratuit este foarte simplă. Există multe companii care oferă astfel de domenii și, în plus, și spațiu de găzduire pentru site.

Pentru numele de domenii oferite gratuit structura adresei URL a site-ului este: [http://www.\[nume companie care asigură domeniul\].com/\[nume site plasat\]](http://www.[nume companie care asigură domeniul].com/[nume site plasat]) sau [http://www.\[nume site plasat\].\[nume companie care asigură domeniul\].com](http://www.[nume site plasat].[nume companie care asigură domeniul].com)

Exemple: <http://sites.google.com/site/scorlat>
 <http://scorlat.blogspot.com>

Pentru site-urile persoanelor juridice (ale companiilor) o asemenea structură este mai puțin acceptabilă. Pentru site-urile educaționale, personale etc. domeniul

gratuit asigurat de o companie care nu plasează în pagini publicitate, precum și utilizarea numelor gratuite este o soluție momentană comodă.

Printre cele mai populare se numără serviciile gratuite oferite de WordPress, Blogspot, Google.

Găzduirea Web (Web hosting)

Dacă a fost selectat un domeniu gratuit, serviciile de oferire a spațiului pentru găzduire fac parte din pachetul de servicii oferit de compania gazdă (de exemplu, Google oferă la moment 100 MB pentru un site, dar permite și crearea mai multor site-uri corelate).

În cazul serviciilor comerciale, există câteva opțiuni de găzduire:

- Instalarea unui server Web propriu;
- Utilizarea serverului Web al furnizorului de servicii Internet;
- Utilizarea unui server Web aparținând unei firme care oferă servicii de hosting.

Server Web propriu

Este cea mai costisitoare soluție, dar oferă maximum de flexibilitate și libertate. Utilizarea serverului propriu necesită procurarea echipamentelor hardware specializate, acces permanent pe bandă largă la rețeaua Internet, servicii speciale oferite de furnizorul de servicii Internet etc. Este soluția optimă pentru companii mari, cu departamente dedicate asigurării serviciilor Web.

Serverul Web al furnizorului de servicii Internet

Utilizarea serverului Web al furnizorului de Internet este o soluție rațională și puțin costisitoare. Mulți dintre furnizorii de servicii Internet oferă în cadrul contractului de prestare a serviciilor și servicii suplimentare de găzduire. Neajunsul soluției constă în limitele impuse de prestator față de spațiul alocat și structura site-ului.

Utilizarea unui server de găzduire

Este o soluție optimă, deoarece permite stabilirea prețului pentru servicii în funcție de spațiul necesar și setul de servicii solicitate. Suplimentar companiile oferă diverse facilități: pagini dinamice, scripturi, baze de date, email personalizat, extinderea dinamică a spațiului alocat etc.

În Moldova asemenea servicii sînt oferite, de exemplu, de MoldData (prin www.host.md).

Organizarea și denumirea fișierelor

Etapă se realizează la momentul programării site-ului, dar buna funcționare a acestuia după publicare depinde de verificarea denumirilor de fișiere și dosare, a amplasării reciproce înainte de transfer.

Recomandări de organizare

E de dorit ca pe calculatorul local fișierele care fac parte din site să fie organizate exact așa cum vor fi organizate pe serverul Web. Serverul Web are un director „rădăcină” (*root*) unde vor fi stocate toate fișierele site-ului. Directorul unde veți stoca fișierele pe calculatorul local va juca rolul directorului „rădăcină” al serverului. În directorul „rădăcină” va fi plasat fișierul care va conține pagina principală a documentului Web.

Pentru site-uri mici toate fișierele pot fi plasate în același director. Pentru structuri mai dezvoltate se recomandă separarea secțiunilor în directoare separate, fiecare secțiune fiind accesată prin propria pagină principală.

Pentru imagini se recomandă crearea unui director aparte în directorul „rădăcină”, unde să fie plasate toate imaginile din site. Calea de la pagina care apelează o imagine la imaginea respectivă trebuie să fie aceeași și pe calculatorul local, și pe server.

Transferul fișierelor

Transferul fișierelor care formează site-ul de pe calculatorul local pe serverul Web este o operațiune relativ simplă, care constă în copierea fișierelor pe server în locația destinată site-ului copiat. Unele dintre companiile de găzduire asigură o aplicație specială destinată transferului fișierelor, dar cea mai utilizată metodă de transfer este prin intermediul unei aplicații client FTP.

Rezumat

În etapa de publicare pe Web a site-ului trebuie parcurși următorii pași:

- Stabilirea unui nume de domeniu pentru site: gratuit sau plătit.
- Găsirea unui serviciu de găzduire convenabil sub raportul servicii/preț.
- Organizarea și denumirea directoarelor și fișierelor în conformitate cu cerințele sistemului serverului gazdă.
- Transferul fișierelor, cea mai folosită metodă de transfer fiind prin FTP.

Analiză

1. Studiul a fost realizat în baza unor date veridice, preluate de pe site-urile întreprinderilor specializate din Republica Moldova.
2. Au fost identificate toate etapele principale de publicare ale unui site creat local.
3. Au fost stabilite diferite posibilități de realizare ale fiecărei etape de publicare.

Concluzii:

1. Site-urile personale, de testare, cu conținuturi educaționale pot fi publicate folosind tehnicile gratuite.

2. Pentru site-urile comerciale se recomandă domeniile personalizate cu plată.
3. Înainte de publicarea site-ului este necesară o ultimă verificare a numelor și amplasării fișierelor.
4. Procedura individuală de publicare depinde de structura și proprietățile site-ului, pe de o parte, și de organizarea serviciilor companiei de găzduire, pe de-alta. Prin urmare, la momentul publicării site-ului trebuie de concretizat condițiile de transfer.

Anexa 1d. Studiul de caz „Algoritmi de calcul al determinanților”

Clasa a XII-a, profil real

Calculul determinanților numerici este o procedură care necesită un număr mare de operații. Indiferent de algoritmul selectat, odată cu creșterea dimensiunii matricei pentru care se calculează determinantul, timpul necesar pentru execuția programului va crește și el (polinomial – pentru algoritmul direct de calcul; exponențial – pentru algoritmul recursiv).

Un alt pericol este determinat de numărul mare de operații de împărțire și înmulțire, efectuate în algoritmul direct. Rezultatele acestora generează erori de rotunjire care, în funcție de tipul de date folosit, pot modifica substanțial rezultatul final.

Pentru a stabili avantajele și dezavantajele fiecărei metode, poate fi propus elevilor următorul set de resurse și sarcini, toate împreună formînd un studiu de caz.

Resurse:

1. *Manualul de informatică pentru clasa a XII-a*, Paragraful „Calculul determinanților”.
2. Programul care calculează valoarea unui determinant numeric folosind doi algoritmi diferiți, realizat într-un limbaj de programare de nivel înalt.
3. Algoritmul pentru determinarea timpului curent de sistem, realizat în formă de funcție.
4. Seturi de date (matrice numerice) pentru care urmează să fie calculate valorile determinanților și analizată funcționarea subprogramelor de calcul al determinanților.
5. Referințe Web la calculatoare on-line, cu ajutorul cărora pot fi verificate rezultatele obținute.

Scopuri:

1. Determinarea avantajelor și dezavantajelor fiecăruia dintre algoritmi descriși în manual, prin verificarea corectitudinii calculelor și a timpului necesar;

2. Determinarea cauzelor posibile de apariție a erorilor și tipul acestora;
3. Elaborarea recomandărilor pentru utilizarea eficientă a algoritmilor de calcul al determinanților;
4. Prezentarea rezultatelor analizei și ale recomandărilor către auditoriu (clasă).

Perioada de realizare: unitatea de învățare „determinanți și sisteme de ecuații liniare”

Durata: 1 săptămână

Ziua 1: Repartizarea resurselor, explicarea sarcinilor, stabilirea modului de comunicare pentru perioada realizării studiului.

Zilele 2 – 7. Consultări online sau directe cu autorii studiului.

Ziua a 8-a. Prezentarea de către autori a studiului pentru auditoriu (clasă).

Model set resurse:

Program:

```

program p002;
uses crt, dos;
type mat=array[1..12,1..12] of real;
var a: mat;
    z,m,n,i,j,l: integer;
    f: text;
    t: real;
function crd( x:mat;t:integer):real; {algoritm recursiv}
    var    i,j,k:integer;
           s: real;
           minor:mat;
    begin if t=1 then crd:=x[1,1]
           else begin s:=0;
                   for k:=1 to t do
                       begin
                           for i:=1 to t-1 do for j:=1 to k-1 do
                               minor[i,j]:=x[i+1,j];
                           for i:=1 to t-1 do for j:=k to t-1 do
                               minor[i,j]:=x[i+1,j+1];
                           if odd(k) then s:=s+x[1,k]*crd(minor, t-1)
                               else s:=s-x[1,k]*crd(minor, t-1);
                           end;
                           crd:=s;
                       end;
                   end;
    end;

```

```

function CID(x:mat;r:integer): real; {algoritm iterativ}
var i,j,k:integer;
    q:real;
begin {CID}
    for i:=1 to r-1 do
        begin
            if x[i,i]=0 then
                begin
                    k:=i;
                    for j:=i+1 to r do
                        if x[j,i]<>0 then k:=j;
                        if k=i then begin CID:=0; exit; end
                        else begin
                            for j:=1 to r do
                                begin q:=x[i,j];
                                x[i,j]:=x[k,j];
                                x[k,j]:=-q;
                                end;
                                end;
                            end;
                            for j:=i+1 to r do
                                begin q:=-x[j,i]/x[i,i];
                                for k:=i to r do x[j,k]:=x[j,k]+x[i,k]*q;
                                end;
                            end;
                        end;
                    end;
                end;
            q:=1;
            for i:=1 to r do q:=q*x[i,i];
            CID:=q;
        end;
begin
assign(f, 'd4.txt'); reset(f);
readln(f,n);
        for i:=1 to n do for j:=1 to n do read(f, a[i,j]); close(f);
    t:=cid(a,n);
    {Aici se va afisa valoarea determinantului, timpul si metoda }
    t:=crd(a,n);
    {Aici se va afisa valoarea determinantului, timpul si metoda }
end.

```

Funcția pentru determinarea timpului curent:

```

function timpcurrent: real;
var h,m,s,ms: word;
begin
    gettime(h,m,s,ms);
    timpcurrent:=h*3600+m*60+s+ms/100;
end;

```

Seturi de date:

Set 1

$$\begin{pmatrix} 6 & 3 & 8 & 2 & 7 & 1 & 7 \\ 2 & 8 & 1 & 0 & 4 & 8 & 9 \\ 9 & 5 & 2 & 8 & 3 & 1 & 0 \\ 8 & 9 & 2 & 4 & 3 & 2 & 5 \\ 6 & 9 & 4 & 6 & 8 & 4 & 9 \\ 2 & 3 & 5 & 6 & 4 & 1 & 7 \\ 0 & 6 & 4 & 2 & 0 & 4 & 8 \end{pmatrix}$$

Set 2

$$\begin{pmatrix} 4 & 6 & 7 & 0 & 0 & 7 & 3 & 7 & 1 \\ 7 & 0 & 6 & 6 & 6 & 3 & 5 & 3 & 4 \\ 0 & 1 & 6 & 6 & 3 & 3 & 5 & 2 & 5 \\ 1 & 5 & 6 & 6 & 2 & 4 & 9 & 5 & 4 \\ 4 & 6 & 2 & 8 & 4 & 8 & 5 & 8 & 3 \\ 7 & 3 & 1 & 1 & 4 & 7 & 6 & 8 & 4 \\ 0 & 8 & 1 & 0 & 1 & 9 & 6 & 2 & 3 \\ 5 & 4 & 4 & 3 & 3 & 5 & 1 & 1 & 4 \\ 3 & 3 & 5 & 1 & 0 & 3 & 9 & 0 & 5 \end{pmatrix}$$

Set 3

$$\begin{pmatrix} 1 & 7 & 5 & 4 & 5 & 3 & 9 & 1 & 7 & 0 \\ 5 & 3 & 3 & 2 & 1 & 7 & 5 & 9 & 1 & 1 \\ 0 & 8 & 9 & 3 & 9 & 2 & 5 & 2 & 0 & 8 \\ 1 & 1 & 6 & 9 & 7 & 8 & 5 & 7 & 3 & 0 \\ 7 & 6 & 7 & 8 & 2 & 8 & 5 & 9 & 5 & 6 \\ 9 & 3 & 2 & 6 & 6 & 0 & 7 & 8 & 7 & 5 \\ 3 & 2 & 2 & 8 & 9 & 5 & 8 & 3 & 2 & 3 \\ 8 & 5 & 3 & 8 & 1 & 9 & 0 & 3 & 7 & 0 \\ 9 & 0 & 9 & 2 & 2 & 1 & 5 & 9 & 9 & 9 \\ 2 & 2 & 0 & 6 & 2 & 7 & 7 & 8 & 2 & 6 \end{pmatrix}$$

Set 4

$$\begin{pmatrix} 0 & -4 & 17 & -7 & 19 & -6 \\ -12 & 12 & -4 & 11 & -15 & 0 \\ 15 & -16 & 19 & -5 & 9 & -16 \\ -12 & 2 & -12 & 3 & -5 & 7 \\ 8 & -14 & 7 & -1 & 1 & -14 \\ -5 & 5 & -16 & 12 & -7 & 16 \end{pmatrix}$$
Rezultatele studiului*Rezultate obținute după integrarea programului și a funcției:*

Setul 1:

```

Turbo Pascal
valoare determinant: -448319.999999000000 timp: 0.0000000000 iterativ
valoare determinant: -2282674.0000000000 timp: 0.0000000000 recursiv

```

Setul 2:

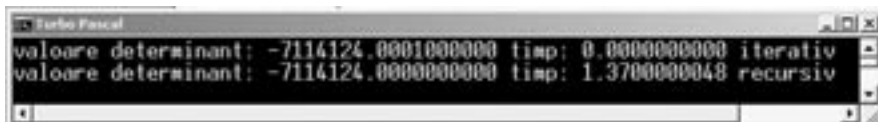
```

Turbo Pascal
valoare determinant: 1155381746.8000000000 timp: 0.0000000000 iterativ
valoare determinant: 1155381747.0000000000 timp: 15.7899999798 recursiv

```

Setul 3:

```
Turbo Pascal
valoare determinant: 138654.0000000000 timp: 0.0000000000 iterativ
valoare determinant: 138654.0000000000 timp: 0.0000000000 recursiv
```

Setul 4:

```
Turbo Pascal
valoare determinant: -7114124.0001000000 timp: 0.0000000000 iterativ
valoare determinant: -7114124.0000000000 timp: 1.3700000048 recursiv
```

Analiză:

1. Prin generarea repetată a unor matrice de ordin nu mai mare decât 10, cu elemente întregi, ce nu depășesc în valoare absolută 20, se poate de stabilit că în majoritatea cazurilor rezultatele obținute prin ambele metode coincid. (exemplu – setul 1 de date)
2. În unele cazuri apar diferențe mici ale rezultatelor, algoritmul recursiv păstrează exactitatea, algoritmul iterativ calculează mai puțin exact. (exemple – seturile de date 2,3)
3. Pentru un număr relativ mic al seturilor generate apar diferențe esențiale ale rezultatelor – erorile sînt generate de algoritmul iterativ, cel recursiv rămîne exact. (setul de date 4)
4. Timpul de lucru al algoritmului recursiv crește exponențial. Pentru matrice de dimensiuni mai mari decât 10 el devine puțin eficient.
5. Toate rezultatele pot fi verificate prin „expertiză independentă” pe www.solveymath.com .

Concluzii:

1. Pentru matrice de ordin mai mic decât 10 se recomandă utilizarea algoritmului recursiv, care este mai exact și nu consumă resurse mari de timp. (Pentru a reduce suplimentar timpul de execuție se poate modifica tipul de date din tipul mat – integer.)
2. Pentru matrice de ordin mai mare este mai rațional de a aplica algoritmul iterativ, dar veridicitatea rezultatelor nu poate fi garantată. Se recomandă reverificarea rezultatelor cu ajutorul unui calculator matematic.

Note:

1. Pentru măsurarea timpului este necesară includerea bibliotecii dos.
2. Rezultatele pot varia în funcție de calculatorul la care se execută programul și de compilatorul folosit.

Bibliografie

1. *Informatică. Curriculum pentru clasele a X-a – a XII-a*, Știința, Chișinău, 2010.
2. *Științe exacte. Matematică. Fizică. Informatică. Curriculum național. Ghid metodologic de implementare pentru învățământul liceal*, C.E. Pro Didactica, Chișinău, 2000.
3. Sergiu Corlat, Lilia Ivanov, *Calculatorul în predare și învățare. Ghid metodologic pentru formarea cadrelor didactice din învățământul preuniversitar*, Știința, Chișinău, 2007.
4. Vasile Andronic, *Informatica. Ghid de implementare a curriculum-ului modernizat în învățământul liceal*, Știința, Chișinău, 2007.
5. Horst Shaub, Karl G. Zenke, *Dicționar de pedagogie*, Polirom, Iași, 2001.
6. Marilyn Fryer, *Predarea și învățarea creativă*, Editura Uniunii Scriitorilor, Chișinău, 2004.
7. Colis, B. and Moonen, J., *Flexible Learning in a Digital World: Experiences and expectations*, London: Kogan-Page, 2001.
8. M. Bocoș, *Teoria și practica cercetării pedagogice*, Editura Casa Cărții de Știință, Cluj-Napoca, 2003.
9. Malcolm Shepherd Knowles, *Self-directed learning: A guide for learners and teachers*, Englewood Cliffs: Prentice Hall/Cambridge, 1975.
10. Wilson, B. G., *Metaphors for instruction: Why we talk about learning environments*. *Educational Technology*, 35 (5), 25-30, (1995).
11. Heather Fry, Steve Ketteridge, Stephanie Marshall, *A handbook for teaching and learning in education: enhancing academic practice*, Taylor & Francis, USA New York, 2009.
12. Ioan Cerghit, *Sisteme de instruire alternative și complementare. Structuri, stiluri, strategii*, Polirom, Iași, 2008.
13. Christiane Bosman, François-Marie Gérard, Xavier Roegiers, *Quel avenir pour les compétences?*, De Boes & Larcier s.a., 2002.
14. *Învățarea centrată pe elev. Ghid pentru profesori și formatori*, Proiectul PHARE: RO. IMC Consulting Ltd, 2005.
15. M. Manolescu, *Evaluarea școlară. Metode, tehnici, instrumente*, Editura Meteor Press, București, 2005.