

## Proiect de tehnologie didactică

Școala : -

Disciplina : Informatică

Clasa : a IX-a

Profilul : Matematică-informatică, intensiv informatică

Data : -

Profesor : -

Unitatea de învățare : Elaborarea algoritmilor de rezolvare a problemelor și implementarea lor într-un limbaj de programare

Tema lecției : Structura repetitivă condiționată anterior

Tipul lecției : Lecție de fixare și formare de deprinderi și priceperi

Durata : 100 de minute

Competențe generale :

(CG1) Identificarea datelor care intervin într-o problemă.

(CG3) Elaborarea algoritmilor de rezolvare a problemelor.

(CG4) Aplicarea algoritmilor fundamentali de prelucrare a datelor.

(CG5) Implementarea algoritmilor într-un limbaj de programare.

Competențe specifice :

(CS3.1) Analizarea enunțului unei probleme și stabilirea pașilor de rezolvare a problemei.

(CS3.2) Reprezentarea algoritmilor în pseudocod.

(CS3.3) Respectarea principiilor programării structurate în procesul de elaborare a algoritmilor.

(CS5.1) Elaborarea unui algoritm de rezolvare a unor probleme din aria curriculară a specializării.

(CS5.2) Alegerea unui algoritm eficient de rezolvare a unei probleme.

Competențe derivate :

La sfârșitul activității didactice, elevii vor fi capabili să :

(CD1) Identifice necesitatea utilizării structurii repetitive condiționate anterior în algoritmul fundamental pentru descompunere în factori primi.

(CD2) Reprezinte în pseudocod algoritmul propus folosind structura *cat timp... execută*.

(CD3) Să argumenteze eficiența algoritmului propus.

(CD4) Să recunoască forme de aplicare a algoritmilor fundamentali discutați în diverse probleme.

C1. Competențe cognitive

La sfârșitul lecției, elevii vor fi capabili :

(C1.1) : să analizeze o problemă, să descrie etapele algoritmului de rezolvare ;

(C1.2) : să identifice tipul structurii repetitive adecvate ;

(C1.3) : să scrie în pseudocod algoritmi eficienți pentru rezolvarea cerințelor propriu zise.

C2. Competențe afective

(C2.1) : să argumenteze corect alegerea structurilor de control și eficiența algoritmului propus ;

(C2.2) : să se autoevalueze corect ;

(C2.3) : să dovedească curiozitate și interes pentru noțiunile utilizate.

C3. Competențe psihomotorii

(C3.1) : să utilizeze corect în problemele propuse algoritmul fundamental de descompunere în factori primi.

C4. Competențe acționale

(C4.1) : să identifice structura de control *cat timp... execută* ;

(C4.2) : să descrie modul de utilizare a algoritmului fundamental în rezolvarea problemelor propuse ;

(C4.3) : să scrie în pseudocod algoritmul corect.

Strategii didactice

Principii didactice :

- principiul legării teoriei de practică ;
- principiul sistematizării și continuității cunoștințelor ;
- principiul accesibilității ;
- principiul individualizării și diferențierii învățării.

Metode și procedee didactice : problematizarea (P), algoritimizarea (A), conversația frontală și individuală (Cv), explicația (E), munca independentă (M), demonstrația (D).

Forme de organizare : lucrul individual și pe grupe de 2-3 elevi

Forme de dirijare a învățării :

- independentă ;
- dirijată de profesor prin mijloacele de învățare.

Metode de evaluare :

- evaluare continuă pe parcursul lecției ;
- apreciere verbală ;
- evaluare practică.

Resurse materiale :

- fișe de aplicații ;
- material bibliografic : culegere de probleme, manual pentru clasa a IX-a, [CȘ], [K1], [CLR], [L].

Structura lecției pe secvențe de instruire

*Ma selgiris*



ID	Min.	Etapale lecției - activitatea elev-profesor	MD
	2'	<b>Moment organizatoric</b> <i>Profesorul:</i> verifică frecvența elevilor, verifică existența resurselor materiale. <i>Elevii:</i> se pregătesc pentru oră.	
	4'	<b>Captarea atenției</b> <i>Profesorul:</i> anunță tema lecției, competențele de format și distribuie fișele de lucru, apoi explică modul de desfășurare a orei. Fișele conțin un număr de sarcini obligatorii, iar pentru elevii cu performanțe mai bune se prevăd câteva sarcini suplimentare. Vezi fișa. <i>Elevii:</i> răspund solicitărilor profesorului, cer lămuriri.	
	5'	<b>Desfășurarea lecției</b> <b>Actualizarea cunoștințelor dobândite în lecțiile anterioare</b> <i>Profesorul</i> solicită elevilor să descrie modul de funcționare a structurii repetitive condiționate anterior <i>cat timp...</i> execută, apoi să explice cum se aplică aceasta în cazul algoritmului fundamental de descompunere în factori primi, punând accent pe modalitățile de eficientizare a timpului de execuție. <i>Elevii:</i> răspund solicitărilor profesorului. Dacă răspunsurile așteptate nu apar în scurt timp, profesorul va numi elevi care să răspundă.	Cv
	60'	<b>Dirijarea învățării</b> Profesorul solicită elevilor să citească cerințele de pe fișa de lucru. Sunt discutate la tablă câteva exemple numerice pentru o mai bună înțelegere a enunțului, evidențiindu-se posibilitățile de optimizare. Profesorul pune întrebări, stimulând prin conversație participarea activă a elevilor la oră. Întrebări posibile: <ul style="list-style-type: none"> <li>• Care este cel mai mic număr care are aceiași factori primi în descompunere ca și <math>n</math>?</li> <li>• Cum putem adapta algoritmul elementar de descompunere în factori primi pentru a determina acest număr?</li> <li>• Cum putem adapta algoritmul elementar de descompunere în factori primi pentru a verifica dacă numărul <math>n</math> este liber de pătrate (nu conține în descompunerea în factori primi niciun factor prim la putere pară)?</li> <li>• Cum putem calcula numărul de divizori ai unui număr natural nenul <math>n</math>, folosind descompunerea în factori primi? Este un algoritm mai eficient decât parcurgerea divizorilor proprii posibili și numărarea celor care sunt divizori ai lui <math>n</math>?</li> <li>• Cum calculăm numărul de cifre 0 în care s-ar termina produsul unor numere naturale citite pe rând până la citirea lui 0?</li> <li>• Putem aplica același algoritm pentru...</li> </ul>	P, Cv

#### Asigurarea transferului

Răspunsurile așteptate ale elevilor:

- algoritmul elementar de descompunere în factori primi (cu parcurgerea optimizată a divizorilor primi posibili până la  $\lfloor \sqrt{n} \rfloor$ );

d-2

cat timp d\*d<n executa

p=0

cat timp n%d=0 executa

n=n/d

p=p+1

sfarsit cat timp

daca p>0 atunci

prelucreaza(d,p)

sfarsit daca

d=d+1

sfarsit cat timp

daca n>1 atunci

prelucreaza(n,1)

sfarsit daca

- cel mai mic număr natural care are aceeași factori primi în descompunere ca și  $n$  se obține făcând produsul factorilor primi distincți ai lui  $n$ , obținuți în urma descompunerii în factori primi a acestuia;

- pentru verificarea unui număr liber de pătrate, *prelucreaza*( $d$ ,  $p$ ) se va înlocui cu verificarea dacă  $p\%2=0$ , caz în care răspunsul este NU;

- pentru numărul divizorilor lui  $n$  se poate aplica formula lui Euler bazată pe descompunerea în factori primi:

dacă  $n=d_1^{p_1} * d_2^{p_2} * d_k^{p_k}$ , unde  $d_1, d_2, \dots, d_k$  sunt factorii primi din descompunerea lui  $n$ , iar  $p_1, p_2, \dots, p_k$  sunt puterile acestora, numărul divizorilor lui  $n$  este  $(p_1+1) * (p_2+1) * \dots * (p_k+1)$ . Acest algoritm are complexitatea timp  $T(n) = O(\sqrt{n})$ , în timp ce algoritmul care parcurge divizorii proprii posibili până la  $\lfloor n/2 \rfloor$  are complexitatea  $T(n) = O(n)$ .

- pentru determinarea numărului de cifre 0 de la sfârșitul produsului numerelor naturale citite până la 0 este necesară calcularea puterii lui 2 și a puterii lui 5 în numerele date, apoi afișarea minimului acestor două puteri, deoarece un 0 de la sfârșitul produsului înseamnă o înmulțire cu  $2*5$ .

E

A

M

E, Cv



D4.	- pentru determinarea numărului de cifre 0 de la sfârșitul lui $n!$ nu se va aplica același algoritm deoarece se observă că în $n!$ este relevantă doar puterea lui 5, care se calculează cu formula $\frac{n}{5} + \frac{n}{5^2} + \frac{n}{5^3} + \dots + \frac{n}{5^k}$ , unde $5k$ este cea mai mare putere a lui 5, mai mică sau egală cu $n$ .	D E
2.2.	<b>Asigurarea intensificării retenției</b> Vor veni la tablă elevi din fiecare echipă, care să exemplifice funcționalitatea fiecărui algoritm propus pe date concrete, pentru o mai bună înțelegere a enunțurilor. Apoi, un alt elev din aceeași echipă va rezolva la tablă și va explica algoritmul propus, după care va argumenta eficiența acestuia.	
1.3.	Elevii notează în caiete algoritmul prezentat, exemplele numerice și argumentele pentru eficiența acestora.	
5'	Profesorul monitorizează activitatea individuală a elevilor, le acordă permanent suport. Dacă au fost identificate greșeli repetate la mai mulți elevi, se interzice lecția pentru o discuție cu toată clasa și lămurirea neclarităților.	
1.3.	Profesorul încurajează elevii cu performanțe superioare să rezolve cerințele suplimentare, insistând asupra eficienței timp a algoritmilor și asupra cazurilor particulare.	
	<b>Evaluarea</b> - pe parcurs, prin aprobare, dezaprobare verbală în urma răspunsurilor date de elevi la întrebări sau în urma observării sistematice a muncii independente; - la începutul orei, profesorul negociază cu clasa câteva criterii de evaluare a algoritmilor propuși pentru rezolvarea problemelor din fișa de lucru (corectitudinea algoritmului propus, eficiența, tratarea cazurilor particulare, modul de redactare a algoritmului în pseudocod); - fiecare echipă va prezenta la tablă una dintre problemele rezolvate, astfel încât în final să fie prezentate toate problemele propuse pe fișa de lucru; - evaluare reciprocă (sunt încurajate echipele care au rezolvat aceeași problemă cu cea prezentată la tablă să evalueze soluția propusă de colegii lor); - la finalul orei, cu note argumentate de rezultatele aplicației realizate.	
5'	<b>Apresiasi activității</b> Profesorul face aprecieri privind performanțele elevilor, recomandări de recuperare a celor care nu au reușit să rezolve sarcinile obligatorii și îi notează pe cei care au fost activi. Profesorul poate cere elevilor să se autoevalueze.	
4'	<b>Tema pentru acasă</b> Profesorul propune tema pentru acasă, iar elevii notează în caiete. Tema pentru acasă ar putea fi una sau două probleme asemănătoare cu cele de pe fișa de lucru și, pentru cei capabili de performanțe superioare, probleme date la olimpiade și concursuri de informatică care utilizează acest algoritm fundamental sau alți algoritmi studiați.	

**Clasa:**  
**Tema lecției:** Structura repetitivă condițională anterioară.

#### Fișă de lucru

1. Să se determine cel mai mic număr natural care are aceiași factori primi în descompunere ca și  $n!$ .
2. Să se verifice dacă  $n$  este liber de pătrate (în descompunerea sa în factori primi, niciun factor nu apare la putere pară). Se va afișa răspunsul corespunzător DA/NU.
3. Să se determine numărul de divizori ai lui  $n$ , folosind parcurgerea divizorilor proprii.
4. Să se determine numărul de divizori ai lui  $n$ , folosind formula lui Euler.
5. Să se determine numărul cifrelor de 0 de la sfârșitul produsului numerelor naturale citite până la citirea numărului 0.
6. Să se determine numărul de cifre 0 de la sfârșitul lui  $n!$ .

#### Tema

- Fie  $n$  un număr natural nenul,  $0 < n < 108$ . Se cere:
1. Să se determine cel mai mare factor prim care apare în descompunerea lui  $n$  la o putere impară, dacă există sau să se afișeze mesajul nu există, în caz contrar.
  2. Să se determine cel mai mic număr natural cu care trebuie înmulțit  $n$  astfel încât produsul obținut să fie pătrat perfect. Să se afișeze descompunerea în factori primi a acestui număr, câte o pereche de forma  $d \cdot p$  (divizor exponent) pe o linie.
  3. (\*\*\*) Se citește  $x$ , număr natural,  $1 < x < 105$ . Să se verifice dacă  $x!$  se divide cu  $n$ .

#### Proiect de tehnologie didactică

**Școala:** -

**Disciplina:** Informatică

**Clasa:** a XI-a

**Profilul:** Matematică-informatică, intensiv informatică

**Data:** -

**Profesor:** -

**Unitatea de învățare:** Grafuri neorientate și arbori cu rădăcină

**Tema lecției:** Grafuri hamiltoniene, euleriene și arbori cu rădăcină

**Tipul lecției:** Lecție de recapitulare-evaluare

**Durată:** 50 de minute

**Competențe generale:**

(CG1) Identificarea datelor care intervin într-o problemă și aplicarea algoritmilor fundamentali de prelucrare a acestora.

(CG2) Elaborarea algoritmilor de rezolvare a problemelor.

(CG3) Implementarea algoritmilor într-un limbaj de programare.

**Competențe specifice:**

(CS1.1) Transpunerea unei probleme din limbaj natural în limbaj de grafuri, folosind corect terminologia specifică.

(CS1.2) Analizarea unei probleme în scopul identificării datelor necesare și alegerea

*Mașalagin*



D4.	D	- pentru determinarea numărului de cifre 0 de la sfârșitul lui $n!$ nu se va aplica același algoritm deoarece se observă că în $n!$ este relevantă doar puterea lui 5, care se calculează cu formula $\frac{n}{5} + \frac{n}{5^2} + \frac{n}{5^3} + \dots + \frac{n}{5^k}$ , unde $5^k$ este cea mai mare putere a lui 5, mai mică sau egală cu $n$ .
2.2.	E	<b>Asigurarea Intensificării retenției</b> Vor veni la tablă elevi din fiecare echipă, care să exemplifice funcționalitatea fiecărui algoritm propus pe date concrete, pentru o mai bună înțelegere a enunțurilor. Apoi, un alt elev din aceeași echipă va rezolva la tablă și va explica algoritmul propus, după care va argumenta eficiența acestuia.
1.3.		Elevii notează în caiete algoritmi prezenți, exemplele numerice și argumentele pentru eficiența acestora. <i>Profesorul</i> monitorizează activitatea individuală a elevilor, le acordă permanent suport. Dacă au fost identificate greșeli repetate la mai mulți elevi, se interrupe lecția pentru o discuție cu toată clasa și lămurirea neclarităților. <i>Profesorul</i> încurajează elevii cu performanțe superioare să rezolve cerințele suplimentare, insistând asupra eficienței timp a algoritmilor și asupra cazurilor particulare.
1.2.		<b>Evaluarea</b> - pe parcurs, prin aprobare, dezaprobare verbală în urma răspunsurilor date de elevi la întrebări sau în urma observării sistematice a muncii independente; - la începutul orei, profesorul negociază cu clasa câteva criterii de evaluare a algoritmilor propuși pentru rezolvarea problemelor din fișa de lucru (corectitudinea algoritmului propus, eficiența, tratarea cazurilor particulare, modul de redactare a algoritmului în pseudocod); - fiecare echipă va prezenta la tablă una dintre problemele rezolvate, astfel încât în final să fie prezentate toate problemele propuse pe fișa de lucru; - evaluare reciprocă (sunt încurajate echipele care au rezolvat aceeași problemă cu cea prezentată la tablă să evalueze soluția propusă de colegii lor); - la finalul orei, cu note argumentate de rezultatele aplicației realizate.
5'		<b>Aprecierea activității</b> Profesorul face aprecieri privind performanțele elevilor, recomandări de recuperare celor care nu au reușit să rezolve sarcinile obligatorii și îi notează pe cei care au fost activi. Profesorul poate cere elevilor să se autoevalueze.
4'		<b>Tema pentru acasă</b> Profesorul propune tema pentru acasă, iar elevii notează în caiete. Tema pentru acasă ar putea fi una sau două probleme asemănătoare cu cele de pe fișa de lucru și, pentru cei capabili de performanțe superioare, probleme date la olimpiade și concursuri de informatică care utilizează acest algoritm fundamental sau alți algoritmi studiați.

Mazalegiu

Clasa:

Tema lecției: Structura repetitivă condiționată anterior.

## Fișă de lucru

Fie  $n$  un număr natural nenul,  $0 < n < 108$ . Se cere:

1. Să se determine cel mai mic număr natural care are aceiași factori primi în descompunerea sa și  $n$ .
2. Să se verifice dacă  $n$  este liber de pătrate (în descompunerea sa în factori primi, niciun factor nu apare la putere pară). Se va afișa răspunsul corespunzător DA/NU.
3. Să se determine numărul de divizori ai lui  $n$ , folosind parcurgerea divizorilor proprii.
4. Să se determine numărul de divizori ai lui  $n$ , folosind formula lui Euler.
5. Să se determine numărul cifrelor de 0 de la sfârșitul produsului numerelor naturale citite până la citirea numărului 0.
6. Să se determine numărul de cifre 0 de la sfârșitul lui  $n!$ .

## Temă

Fie  $n$  un număr natural nenul,  $0 < n < 108$ . Se cere:

1. Să se determine cel mai mare factor prim care apare în descompunerea lui  $n$  la o putere impară, dacă există sau să se afișeze mesajul *nu există*, în caz contrar.
2. Să se determine cel mai mic număr natural cu care trebuie înmulțit  $n$  astfel încât produsul obținut să fie pătrat perfect. Să se afișeze descompunerea în factori primi a acestui număr, câte o pereche de forma  $d \cdot p$  (divizor exponent) pe o linie.
3. (\*\*). Se citește  $x$ , număr natural,  $1 < x < 105$ . Să se verifice dacă  $x!$  se divide cu  $n$ .

## Proiect de tehnologie didactică

Școala: -

Disciplina: Informatică

Clasa: a XI-a

Profilul: Matematică-informatică, intensiv informatică

Data: -

Profesor: -

Unitatea de învățare: Grafuri neorientate și arbori cu rădăcină

Tema lecției: Grafuri hamiltoniene, euleriene și arbori cu rădăcină

Tipul lecției: Lecție de recapitulare-evaluare

Durată: 50 de minute

Competențe generale:

(CG1) Identificarea datelor care intervin într-o problemă și aplicarea algoritmilor fundamentali de prelucrare a acestora.

(CG2) Elaborarea algoritmilor de rezolvare a problemelor.

(CG3) Implementarea algoritmilor într-un limbaj de programare.

Competențe specifice:

(CS1.1) Transpunerea unei probleme din limbaj natural în limbaj de grafuri, folosind corect terminologia specifică.

(CS1.2) Analizarea unei probleme în scopul identificării datelor necesare și alegerea