



# ROMÂNIA

MINISTERUL EDUCAȚIEI, CERCETĂRII ȘI INOVĂRII  
Centrul Național pentru Curriculum și Evaluare în Învățământul Preuniversitar

Str. General Berthelot nr. 26, sector 1, București, 010168,  
Tel.: +40-21-3144411; 3144511; 3144424. Tel/fax: +40-21-3103207

## Evaluarea la disciplina *Informatică*

### În cadrul examenului național de bacalaureat 2010

Disciplina Informatică are statutul de disciplină opțională la proba E – d). Pentru această disciplină există șase tipuri diferite de subiecte, în funcție de specializare și limbajul studiat:

1. Informatică – pentru specializarea matematică informatică, intensiv informatică - limbaj Pascal ;
2. Informatică – pentru specializarea matematică informatică, intensiv informatică - limbaj C/C++;
3. Informatică – pentru specializarea matematică informatică -limbaj Pascal ;
4. Informatică – pentru specializarea matematică informatică -limbaj C/C++;
5. Informatică – pentru specializarea științe ale naturii -limbaj Pascal ;
6. Informatică – pentru specializarea științe ale naturii -limbaj C/C++.

Pentru fiecare dintre aceste tipuri de subiecte s-au elaborat câte trei grupe de itemi (I, II, III), fiecare grupă având cinci itemi, de dificultate diferită:

- Foarte ușor – 4 puncte
- Ușor – 6 puncte
- Mediu – 10 puncte
- Dificil – 6 puncte
- Foarte dificil - 4 puncte

#### Tipuri de itemi utilizați:

pentru grupa I:

1. item obiectiv cu alegere multiplă
2. item semiobiectiv – întrebări structurate

pentru grupa II:

1. itemi obiectivi cu alegere multiplă
2. itemi semiobiectivi – întrebări cu răspuns scurt și itemi de completare
3. item subiectiv – rezolvare de probleme

pentru grupa III:

1. item obiectiv cu alegere multiplă
2. item semiobiectiv – întrebări cu răspuns scurt și itemi de completare
3. itemi subiectivi – rezolvare de probleme

Competențele și conținuturile menționate atât în programa școlară, cât și în programa de bacalaureat pentru disciplina informatică pot fi puse în valoare prin oricare din limbajele de programare Pascal, respectiv C/C++. De aceea cerințele pentru cele două modele, corespunzătoare celor două limbaje, sunt comune, dar limbajul de implementare / exemplificare este diferit. La conceperea itemilor s-a avut în vedere ca rezolvarea acestora să aibă același grad de dificultate pentru ambele limbaje menționate.

## A) specializarea matematică – informatică intensiv informatică

**Competențele** menționate în programele de bacalaureat pentru disciplina Informatică

sunt:

- C1. construirea algoritmilor corespunzători unor prelucrări elementare și reprezentarea lor prin intermediul programelor pseudocod și programelor scrise în limbaj de programare (Pascal sau C/C++, la alegere);
- C2. analiza rezolvării unei probleme prin urmărirea evoluției valorilor variabilelor prelucrate de algoritmul corespunzător;
- C3. abstractizarea rezolvării prin construirea unor algoritmi echivalenți;
- C4. identificarea și utilizarea tipurilor de date predefinite specifice unui limbaj de programare;
- C5. definirea și utilizarea unor tipuri de date proprii;
- C6. identificarea și utilizarea operatorilor predefiniți elementari;
- C7. identificarea și utilizarea subprogramelor predefinite elementare;
- C8. identificarea și utilizarea regulilor sintactice specifice limbajului de programare studiat;
- C9. definirea și apelul unor subprograme proprii cu înțelegerea mecanismelor de transfer prin intermediul parametrilor;
- C10. identificarea proprietăților unor structuri de date necesare în rezolvarea problemelor cu ajutorul calculatorului și utilizarea unor modele de memorare a acestora;
- C11. organizarea datelor ce intervin în rezolvarea unei probleme utilizând structuri de date adecvate;
- C12. organizarea etapelor de prelucrare ce formează un algoritm utilizând structuri de control și module de program;
- C13. folosirea unor metode sistematice de rezolvare pentru probleme de generare;
- C14. analiza unor algoritmi echivalenți de rezolvare a unei probleme în vederea alegerii algoritmului optim.

Distribuirea competențelor pe grupele de itemi este:

Grupa de itemi	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
I														
II														
III														

**Domeniile de conținuturi** prin care se evaluează competențele propuse sunt:

- Algoritmi - pseudocod
- Elementele de bază ale unui limbaj de programare (Pascal sau C, la alegere)
- Subprograme predefinite
- Tipuri structurate de date
- Fișiere text
- Algoritmi elementari
- Subprograme definite de utilizator
- Recursivitate
- Metoda backtracking (iterativă sau recursivă)
- Generarea elementelor combinatoriale
- Structuri dinamice de date (alocare dinamică)
- Grafuri

Distribuirea domeniilor de conținuturi pe grupe de itemi este:

Nr.crt	Domeniu de conținut	I	II	III
1.	Algoritmi - pseudocod			
2.	Elementele de bază ale unui limbaj de programare			
3.	Subprograme predefinite			
4.	Tipuri structurate de date: - tablouri bidimensionale - șiruri de caractere - înregistrări			
	Tipuri structurate de date: - tablouri unidimensionale			
5.	Fișiere text			
6.	Algoritmi elementari			
7.	Subprograme definite de utilizator			
8.	Recursivitate			
9.	Metoda backtracking (iterativă sau recursivă)			
10.	Generarea elementelor combinatoriale			
11.	Structuri dinamice de date (alocare dinamică)			
12.	Grafuri			

## B) specializarea matematică – informatică

- C1. construirea algoritmilor corespunzatori unor prelucrări elementare și reprezentarea lor prin intermediul programelor pseudocod și programelor scrise în limbaj de programare (Pascal sau C/C++, la alegere);
- C2. analiza rezolvării unei probleme prin urmărirea evoluției valorilor variabilelor prelucrate de algoritmul corespunzător;
- C3. abstractizarea rezolvării prin construirea unor algoritmi echivalenți;
- C4. identificarea și utilizarea tipurilor de date predefinite specifice unui limbaj de programare;
- C5. definirea și utilizarea unor tipuri de date proprii;
- C6. identificarea și utilizarea operatorilor predefiniți elementari;
- C7. identificarea și utilizarea subprogramelor predefinite elementare;
- C8. identificarea și utilizarea regulilor sintactice specifice limbajului de programare studiat;
- C9. definirea și apelul unor subprograme proprii cu înțelegerea mecanismelor de transfer prin intermediul parametrilor;
- C10. identificarea proprietăților unor structuri de date necesare în rezolvarea problemelor cu ajutorul calculatorului și utilizarea unor modele de memorare a acestora;
- C11. organizarea datelor ce intervin în rezolvarea unei probleme utilizând structuri de date adecvate;
- C12. organizarea etapelor de prelucrare ce formează un algoritm utilizând structuri de control și module de program;
- C13. folosirea unor metode sistematice de rezolvare pentru probleme de generare;
- C14. analiza unor algoritmi echivalenți de rezolvare a unei probleme în vederea alegerii algoritmului optim.

Distribuirea competențelor pe grupele de itemi este:

Grupa de itemi	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
I														
II														
III														

**Domeniile de conținuturi** prin care se evaluează competențele propuse sunt:

- Algoritmi - pseudocod
- Elementele de bază ale unui limbaj de programare (Pascal sau C, la alegere)
- Subprograme predefinite
- Tipuri structurate de date
- Fișiere text
- Algoritmi elementari
- Subprograme definite de utilizator
- Recursivitate
- Metoda backtracking (iterativă sau recursivă)
- Generarea elementelor combinatoriale
- Liste
- Grafuri

Distribuirea domeniilor de conținuturi pe grupe de itemi este:

Nr.crt	Domeniu de conținut	I	II	III
1.	Algoritmi - pseudocod			
2.	Elementele de bază ale unui limbaj de programare			
3.	Subprograme predefinite			
4.	Tipuri structurate de date: - tablouri bidimensionale - șiruri de caractere - înregistrări			
	Tipuri structurate de date: - tablouri unidimensionale			
5.	Fișiere text			
6.	Algoritmi elementari			
7.	Subprograme definite de utilizator			
8.	Recursivitate			
9.	Metoda backtracking (iterativă sau recursivă)			
10.	Generarea elementelor combinatoriale			
11.	Liste			
12.	Grafuri			

### C) specializarea științe ale naturii

- C1. construirea algoritmilor corespunzători unor prelucrări elementare și reprezentarea lor prin intermediul programelor pseudocod și programelor scrise în limbaj de programare (Pascal sau C/C++, la alegere);
- C2. analiza rezolvării unei probleme prin urmărirea evoluției valorilor variabilelor prelucrate de algoritmul corespunzător;
- C3. abstractizarea rezolvării prin construirea unor algoritmi echivalenți;
- C4. identificarea și utilizarea tipurilor de date predefinite specifice unui limbaj de programare;
- C5. definirea și utilizarea unor tipuri de date proprii;
- C6. identificarea și utilizarea operatorilor predefiniți elementari;
- C7. identificarea și utilizarea subprogramelor predefinite elementare;
- C8. identificarea și utilizarea regulilor sintactice specifice limbajului de programare studiat;
- C9. identificarea proprietăților unor structuri de date necesare în rezolvarea problemelor cu ajutorul calculatorului și utilizarea unor modele de memorare a acestora;
- C10. organizarea datelor ce intervin în rezolvarea unei probleme utilizând structuri de date adecvate;
- C11. organizarea etapelor de prelucrare ce formează un algoritm utilizând structuri de control;
- C12. analiza unor algoritmi echivalenți de rezolvare a unei probleme în vederea alegerii algoritmului optim.

Distribuirea competențelor pe grupele de itemi este:

Grupa de itemi	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
I												
II												
III												

**Domeniile de conținuturi** prin care se evaluează competențele propuse sunt:

Algoritmi - pseudocod

Elementele de bază ale unui limbaj de programare (Pascal sau C, la alegere)

Subprograme predefinite

Tipuri structurate de date

Fișiere text

Algoritmi elementari

Distribuirea domeniilor de conținuturi pe grupe de itemi este:

Nr.crt	Domeniu de conținut	I	II	III
1.	Algoritmi - pseudocod			
2.	Elementele de bază ale unui limbaj de programare			
3.	Subprograme predefinite			
4.	Tipuri structurate de date: - tablouri bidimensionale			
5.	Fișiere text			
6.	Algoritmi elementari			

### **Precizări referitoare la evaluarea probei scrise**

În cadrul examenului de bacalaureat evaluarea se realizează prin raportare la competențele de evaluat prezentate în programa disciplinei.

La baza construirii competențelor de evaluat s-au avut în vedere categoriile: cunoaștere; comprehensiune sau înțelegere; aplicare; analiză; sinteză; evaluare.

**1. Cunoașterea** vizează: identificarea de termeni, relații, procese, observarea unor fenomene, procese, nominalizarea unor concepte, culegerea de date din surse variate, definirea unor concepte.

**2. Înțelegerea** vizează: compararea unor date, stabilirea unor relații, calcularea unor rezultate parțiale, clasificări de date, reprezentarea unor date, sortarea-discriminarea, investigarea, descoperirea, explorarea

**3. Aplicarea** vizează: reducerea la o schemă sau model, anticiparea unor rezultate, reprezentarea datelor, remarcarea unor invarianți, rezolvarea de probleme prin modelare și algoritmizare.

**4. Analiza** vizează: descrierea unor stări, sisteme, procese, fenomene, generarea de idei, argumentarea unor enunțuri, demonstrarea, compararea unor rezultate.

**5. Sinteza** vizează: formularea unor concluzii, calcularea și evaluarea unor rezultate, interpretarea rezultatelor, analiza de situații, elaborarea de strategii, relaționări între diferite tipuri de reprezentări.

**6. Evaluarea** vizează: aplicarea, generalizarea și particularizarea, integrarea, verificarea, optimizarea, transpunerea, realizarea de conexiuni, adaptare și adecvare la context.

**Baremul de evaluare și de notare** este instrumentul pe baza căruia se apreciază lucrările elevilor. Este un instrument de evaluare și de notare asociat unei/unor sarcini concrete de lucru date elevilor.

Baremul de evaluare și de notare este elaborat cu un grad înalt de obiectivitate și aplicabilitate, astfel încât să reducă la minimum diferențele de notare dintre corectori.

Baremul de evaluare și notare este proiectat pe baza notării analitice. Aceasta implică determinarea principalelor performanțe (unități de răspuns) pe care elevul trebuie să le evidențieze în răspunsul său la fiecare item. Unităților de răspuns li se acordă puncte care, însumate, determină nota pentru fiecare item. Notarea analitică are avantajul de a asigura rigurozitatea corectării, favorizând realizarea unei aprecieri obiective.

Baremul de evaluare și de notare permite evaluarea precisă a răspunsurilor la itemii de tip alegere multiplă. În cazul itemilor de tip rezolvare de probleme, baremul de corectare și de notare include elemente ale răspunsului care vor fi punctate. În acest fel candidatul primește punctaj pentru rezolvări parțiale ale cerinței itemului. Se vor puncta însă corespunzător oricare alte metode de rezolvare corectă a cerinței.

Baremele realizate sunt comune pentru limbajele C/C++ și Pascal.

Pentru exemplificare, sunt propuse modele de subiect și barem pentru cele șase tipuri de subiecte.

**Examenul de bacalaureat 2010**  
**PROBA E – d)**  
**Proba scrisă la INFORMATICĂ, limbajul Pascal**  
**Specializarea Matematică-informatică intensiv informatică**

**MODEL**

- ◆ Toate subiectele (I, II și III) sunt obligatorii. Se acordă 10 puncte din oficiu.
- ◆ Timpul efectiv de lucru este de 3 ore.
- ◆ În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).

**Subiectul I**

**(30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Care este numărul total de atribuiri efectuate la executarea secvenței de instrucțiuni alăturate? (4p.)
- ```
x:=4; y:=6;  
while y=6 do  
  y:=y+1;  
if x=y then  
  x:=x+1;
```
- a. 4                      b. 3                      c. 2                      d. 5

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

**2. Se consideră algoritmul alăturat descris în pseudocod:**

S-a notat cu  $x\%y$  restul împărțirii numărului natural  $x$  la numărul natural nenul  $y$  și cu  $[z]$  partea întregă a numărului real  $z$ .

- a) Scrieți numărul afișat dacă se citește valoarea  $n=12939$ . (6p.)
- b) Scrieți toate valorile care pot fi citite pentru variabila  $n$  astfel încât, în urma executării algoritmului, valoarea afișată să fie **2009**. (4p.)

```
citește n (număr natural)  
nr ← 0  
p ← 1  
cât timp n ≠ 0 execută  
  c ← n%10  
  dacă c > 0 și c < 9 atunci  
    c ← c+1  
  ■  
  nr ← nr+c*p  
  p ← p*10  
  n ← [n/10]  
  ■  
scrie nr
```

- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura **cât timp...execută** cu o structură repetitivă de un alt tip. (6p.)
- d) Scrieți programul **Pascal** corespunzător algoritmului dat. (10p.)



**Subiectul al II-lea**

**(30 de puncte)**

**Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.**

1. Se consideră un graf orientat cu **6** noduri numerotate de la **1** la **6** și cu mulțimea arcelor formată **doar** din arcele:
- de la fiecare nod numerotat cu un număr neprim **i** ( **$i > 1$** ) la toate nodurile numerotate cu numere ce aparțin mulțimii divizorilor proprii ai lui **i** (divizori diferiți de **1** și de **i**)
  - de la nodul numerotat cu **1** la nodul numerotat cu **6**
  - de la fiecare nod numerotat cu un număr prim **i** la nodul numerotat cu **i-1**
- Pentru graful dat, care este lungimea celui mai mare drum, format **doar** din noduri distincte? **(4p.)**
- a. **6**                                b. **5**                                c. **3**                                d. **4**
2. Câte frunze are arborele cu rădăcină descris prin următorul vector "de tați": **(6,5,5,2,0,3,3,3,8,7,7)?** **(4p.)**
- a. **1**                                b. **2**                                c. **5**                                d. **4**

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

3. În declarația alăturată, câmpurile **x** și **y** ale înregistrării pot memora numărătorul, respectiv numitorul unei fracții. Scrieți secvența de instrucțiuni prin executarea căreia se construiește în variabila **f** o fracție obținută prin însumarea fracțiilor memorate în variabilele **f1** și **f2**. **(6p.)**
- |                                                                                     |                                                                                     |
|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <pre>type fracție =record<br/>  x,y:integer<br/>end;<br/>var f,f1,f2:fracție;</pre> | <pre>type fracție =record<br/>  x,y:integer<br/>end;<br/>var f,f1,f2:fracție;</pre> |
|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
4. În secvența de instrucțiuni de mai jos, variabila **s** memorează un șir de caractere format doar din litere ale alfabetului englez, iar variabila **i** este de tip **integer**.
- Știind că în urma executării secvenței s-a afișat succesiunea de caractere **eied\*ea el\*** scrieți care este șirul de caractere memorat de variabila **s**. **(6p.)**
- |                                                                                                                   |                                                                                                                   |
|-------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <pre>for i:=1 to length(s) do<br/>  if s[i]='e' then<br/>    write('*')<br/>  else<br/>    write('e',s[i]);</pre> | <pre>for i:=1 to length(s) do<br/>  if s[i]='e' then<br/>    write('*')<br/>  else<br/>    write('e',s[i]);</pre> |
|-------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
5. Scrieți un program **Pascal** care citește de la tastatură un număr natural **n** ( **$2 \leq n \leq 24$** ) și construiește în memorie o matrice cu **n** linii și **n** coloane ale cărei elemente vor primi valori după cum urmează:
- elementele aflate pe diagonala principală a matricei vor primi valoarea **0**
  - elementele de pe prima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n**
  - elementele de pe a doua coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n-1**
  - ...
  - elementele de pe ultima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **1**
- Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).
- |                                                                                                                                                                                                    |                                                        |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| <pre>Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei<br/>pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în<br/>exemplu).</pre> | <pre>0 3 2 1<br/>4 0 2 1<br/>4 3 0 1<br/>4 3 2 0</pre> |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
- Exemplu:** pentru **n=4** se va afișa matricea alăturată. **(10p.)**

**Subiectul al III-lea**

**(30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. În timpul procesului de generare a permutărilor mulțimii  $\{1, 2, \dots, n\}$  prin metoda backtracking, în tabloul unidimensional  $x$  este plasat un element  $x_k$  ( $2 \leq k \leq n$ ). Acesta este considerat valid dacă este îndeplinită condiția: **(6p.)**
- a.  $x_k \notin \{x_1, x_2, \dots, x_{k-1}\}$
  - b.  $x_k \neq x_{k-1}$
  - c.  $x_k \notin \{x_1, x_2, \dots, x_n\}$
  - d.  $x_k \neq x_{k-1}$  și  $x_k \neq x_{k+1}$

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Considerăm subprogramul recursiv definit alăturat. Ce se va afișa în urma apelului de mai jos?  
 $f('C')$  **(4p.)**
- ```
procedure f(c:char);  
begin  
  if c > 'A' then f(pred(c));  
  write(c);  
  if c > 'A' then f(pred(c))  
end;
```
3. **a)** Scrieți definiția completă a unui subprogram, **nz**, cu un parametru întreg  $n$  ( $0 < n \leq 32000$ ), care returnează numărul zerourilor de la sfârșitul numărului  $n! = 1 \cdot 2 \cdot \dots \cdot n$ . **(6p.)**  
**b)** Scrieți programul **Pascal** care citește de la tastatură un număr natural  $k$  ( $0 < k \leq 1500$ ) și determină, folosind apeluri ale subprogramului **nz**, cel mai mic număr natural  $n$  pentru care  $n!$  are cel puțin  $k$  zerouri la sfârșit. Numărul determinat se afișează pe ecran. **(4p.)**
4. Scrieți programul **Pascal** care citește din fișierul text **BAC.TXT** numărul întreg  $n$  ( $1 \leq n \leq 10000$ ) și un șir de  $n$  perechi de numere întregi  $a$   $b$  ( $1 \leq a \leq b \leq 32000$ ), fiecare pereche fiind scrisă pe o linie nouă a fișierului, cu un spațiu între cele două numere. Programul afișează pe ecran pentru fiecare pereche  $a, b$  cel mai mare număr natural din intervalul închis  $[a, b]$  care este o putere a lui **2** sau numărul **0** dacă nu există nicio putere a lui **2** în intervalul respectiv. Numerele afișate pe ecran se scriu în linie, separate prin câte un spațiu. Un număr  $p$  este putere a lui **2** dacă există un număr natural  $k$  astfel încât  $p = 2^k$ .  
**Exemplu:** dacă fișierul **BAC.TXT** conține numerele  
**3**  
**2 69**  
**10 20**  
**19 25**  
se va afișa: **64 16 0**. **(10p.)**

**Examenul de bacalaureat 2010**  
**PROBA E - d)**  
**Proba scrisă la INFORMATICĂ, limbajul C/C++**  
**Specializarea Matematică-informatică intensiv informatică**

**MODEL**

- ◆ Toate subiectele (I, II și III) sunt obligatorii. Se acordă 10 puncte din oficiu.
- ◆ Timpul efectiv de lucru este de 3 ore.
- ◆ În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).

**Subiectul I** **(30 de puncte)**

Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Care este numărul total de atribuiri efectuate la executarea  
secvenței de instrucțiuni alăturate? (4p.)
- |             |             |             |             |
|-------------|-------------|-------------|-------------|
| <b>a. 4</b> | <b>b. 3</b> | <b>c. 2</b> | <b>d. 5</b> |
|-------------|-------------|-------------|-------------|
- x=4; y=6;**  
**while (y==6) y=y+1;**  
**if (x==y) x=x+1;**

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Se consideră algoritmul alăturat descris în pseudocod.
- S-a notat cu  $x\%y$  restul împărțirii numărului natural  $x$  la numărul natural nenul  $y$  și cu  $[z]$  partea întregă a numărului real  $z$ .
- a) Scrieți numărul afișat dacă se citește valoarea **n=12939**. (6p.)
- b) Scrieți toate valorile care pot fi citite pentru variabila **n** astfel încât, în urma executării algoritmului, valoarea afișată să fie **2009**. (4p.)
- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura **cât timp...execută** cu o structură repetitivă de un alt tip. (6p.)
- d) Scrieți programul **C/C++** corespunzător algoritmului dat. (10p.)
- citește n (număr natural)  
**nr** ← 0  
**p** ← 1  
cât timp **n** ≠ 0 execută  
  **c** ← n%10  
  dacă **c** > 0 și **c** < 9 atunci  
    **c** ← c+1  
  ■  
  **nr** ← nr+c\*p  
  **p** ← p\*10  
  **n** ← [n/10]  
  ■  
scrie nr

**Subiectul al II-lea** **(30 de puncte)**

**Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.**

1. Se consideră un graf orientat cu 6 noduri numerotate de la 1 la 6 și cu mulțimea arcelor formată **doar** din arcele:
- de la fiecare nod numerotat cu un număr neprim  $i$  ( $i > 1$ ) la toate nodurile numerotate cu numere ce aparțin mulțimii divizorilor proprii ai lui  $i$  (divizori diferiți de 1 și de  $i$ )
  - de la nodul numerotat cu 1 la nodul numerotat cu 6
  - de la fiecare nod numerotat cu un număr prim  $i$  la nodul numerotat cu  $i-1$
- Pentru graful dat, care este lungimea celui mai mare drum, format **doar** din noduri distincte? **(4p.)**
- a. 6                                      b. 5                                      c. 3                                      d. 4
2. Câte frunze are arborele cu rădăcină descris prin următorul vector "de tați": **(4p.)**  
**(6,5,5,2,0,3,3,3,8,7,7)?**
- a. 1                                      b. 2                                      c. 5                                      d. 4

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

3. În declarația alăturată, câmpurile  $x$  și  $y$  ale înregistrării pot memora număratorul, respectiv numitorul unei fracții. Scrieți secvența de instrucțiuni prin executarea căreia se construiește în variabila  $f$  o fracție obținută prin însumarea fracțiilor memorate în variabilele  $f1$  și  $f2$ . **(6p.)**
- |   |  |
|---|--|
| <pre>//C n=strlen(s); for(i=0;i&lt;n;i++)   if (s[i]!='e') printf("%c",s[i]);   else printf("%c%c",'e',s[i]);</pre> | <pre>struct fracție {   int x,y; }f,f1,f2;</pre> |
|---|--|
4. În secvența de instrucțiuni de mai jos, variabila  $s$  memorează un șir de caractere format doar din litere ale alfabetului englez, iar variabilele  $i$  și  $n$  sunt de tip  $int$ . Știind că în urma executării secvenței s-a afișat succesiunea de caractere **eied\*eael\*** scrieți care este șirul de caractere memorat de variabila  $s$ . **(6p.)**
- |   |   |
|---|---|
| <pre>//C n=strlen(s); for(i=0;i&lt;n;i++)   if (s[i]!='e') printf("%c",s[i]);   else printf("%c%c",'e',s[i]);</pre> | <pre>//C++ n=strlen(s); for(i=0;i&lt;n;i++)   if (s[i]!='e') cout&lt;&lt;" ";   else cout&lt;&lt;"e"&lt;&lt;s[i];</pre> |
|---|---|
5. Scrieți un program **C/C++** care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 24$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane ale cărei elemente vor primi valori după cum urmează:
- elementele aflate pe diagonala principală a matricei vor primi valoarea **0**
  - elementele de pe prima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea  $n$
  - elementele de pe a doua coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea  $n-1$
  - ...
  - elementele de pe ultima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **1**
- Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).
- |  |  |
|--|--|
| <pre>Exemplu: pentru n=4 se va afișa matricea alăturată.</pre> | <pre>0 3 2 1 4 0 2 1 4 3 0 1 4 3 2 0</pre> |
|--|--|
- (10p.)**

**Subiectul al III-lea** **(30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. În timpul procesului de generare a permutărilor mulțimii  $\{1, 2, \dots, n\}$  prin metoda backtracking, în tabloul unidimensional  $x$  este plasat un element  $x_k$  ( $2 \leq k \leq n$ ). Acesta este considerat valid dacă este îndeplinită condiția: **(6p.)**
- a.  $x_k \notin \{x_1, x_2, \dots, x_{k-1}\}$  b.  $x_k \neq x_{k-1}$   
c.  $x_k \notin \{x_1, x_2, \dots, x_n\}$  d.  $x_k \neq x_{k-1}$  și  $x_k \neq x_{k+1}$

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Considerăm subprogramul recursiv definit alăturat. Ce se va afișa în urma apelului de mai jos?  
`f('C');` **(4p.)**
- |  |  |
|--|--|
|  | <pre>void f(char c) { if (c&gt;'A') f(c-1);   cout&lt;&lt;c;   printf("%c",c);   if (c&gt;'A') f(c-1); }</pre> |
|--|--|
3. **a)** Scrieți definiția completă a unui subprogram, **nz**, cu un parametru întreg **n** ( $0 < n \leq 32000$ ), care returnează numărul zerourilor de la sfârșitul numărului  $n! = 1 \cdot 2 \cdot \dots \cdot n$ . **(6p.)**  
**b)** Scrieți programul **C/C++** care citește de la tastatură un număr natural **k** ( $0 < k \leq 1500$ ) și determină, folosind apeluri ale subprogramului **nz**, cel mai mic număr natural **n** pentru care **n!** are cel puțin **k** zerouri la sfârșit. Numărul determinat se afișează pe ecran. **(4p.)**
4. Scrieți programul **C/C++** care citește din fișierul text **BAC.TXT** numărul întreg **n** ( $1 \leq n \leq 10000$ ) și un șir de **n** perechi de numere întregi **a b** ( $1 \leq a \leq b \leq 32000$ ), fiecare pereche fiind scrisă pe o linie nouă a fișierului, cu un spațiu între cele două numere. Programul afișează pe ecran pentru fiecare pereche **a, b** cel mai mare număr natural din intervalul închis **[a, b]** care este o putere a lui **2** sau numărul **0** dacă nu există nicio putere a lui **2** în intervalul respectiv. Numerele afișate pe ecran se scriu în linie, separate prin câte un spațiu. Un număr **p** este putere a lui **2** dacă există un număr natural **k** astfel încât  $p = 2^k$ .  
**Exemplu:** dacă fișierul **BAC.TXT** conține numerele  
**3**  
**2 69**  
**10 20**  
**19 25**  
se va afișa: **64 16 0**. **(10p.)**

**Examenul de bacalaureat 2010**  
**Proba E - d)**  
**Proba scrisă la INFORMATICĂ**  
**specializarea matematică-informatică, intensiv informatică**

**BAREM DE EVALUARE ȘI DE NOTARE**  
**(comun pentru limbajele Pascal și C/C++)**

**MODEL**

- Se punctează oricare alte formulări/ modalități de rezolvare corectă a cerințelor.
- Nu se acordă punctaje intermediare, altele decât cele precizate explicit prin barem. Nu se acordă fracțiuni de punct.
- Se acordă 10 puncte din oficiu. Nota finală se calculează prin împărțirea punctajului total acordat pentru lucrare la 10.
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.
- Utilizarea unui tip de date care depășește domeniul de valori precizat în enunț este acceptată dacă acest lucru nu afectează corectitudinea în funcționarea programului.

**SUBIECTUL I**

**(30 de puncte)**

1.	b	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
2.	a) 23949	6 p.	
	b) 1008, 1009 *	4 p.	(*) Pentru fiecare număr corect se acordă 2p
	c) Pentru program pseudocod corect - structură repetitivă corectă * - echivalența prelucrării realizate - algoritm complet - corectitudine globală	6 p. 2 p. 2 p. 1 p. 1 p.	(*) Se va puncta orice formă corectă de structură repetitivă de alt tip (de exemplu execută...cât timp, repetă...până când etc.)
	d) Pentru program corect -declarația corectă a tuturor variabilelor -citire și scriere corecte -structură repetitivă cu test inițial corectă -structură de decizie corectă -atribuiri corecte -corectitudinea globală a programului <sup>1)</sup>	10 p. 2 p. 2 p. 2 p. 2 p. 1 p. 1 p.	Elevii nu vor fi depunțați pentru modul de aliniere al instrucțiunilor.

**SUBIECTUL al II-lea**

**(30 de puncte)**

1)	b	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
2)	c	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.

<p><b>3) Pentru rezolvare corectă*</b></p> <ul style="list-style-type: none"> <li>- accesul corect la câmpurile înregistrării</li> <li>- calculul corect al numărătorului **</li> <li>- calculul corect al numitorului **</li> <li>- corectitudinea sintactică a secvenței</li> </ul>	<p><b>6 p.</b></p> <p>2 p.</p> <p>2 p.</p> <p>1 p.</p> <p>1 p.</p>	<p>(*) O posibilă soluție constă în transcrierea în limbaj de programare a instrucțiunilor de mai jos:</p> <p><b>f.x←f1.x*f2.y+f1.y*f2.x</b></p> <p><b>f.y←f1.y*f2.y</b></p> <p>Se acordă punctajul maxim chiar dacă se fac prelucrări suplimentare (de exemplu simplificări, etc.) care nu denaturează rezultatul, în raport cu cerința.</p> <p>(**) Se acordă numai 1 din 3 p.dacă formulele sunt corecte, dar se confundă termenii numitor / numărător.</p>
<p><b>4) Răspuns corect: ideale</b></p>	<p><b>6 p.</b></p>	<p>Dacă răspunsul diferă printr-o singură literă de răspunsul corect, se acordă numai 3 p.</p>
<p><b>5) Pentru program se acordă punctaj maxim</b></p> <ul style="list-style-type: none"> <li>- declararea corectă a tuturor variabilelor (matrice și variabile simple)</li> <li>- citire corectă</li> <li>- completarea diagonalei cu 0 *</li> <li>-completarea matricei cu valorile nenule impuse **</li> <li>- afișarea matricei în formatul cerut</li> <li>- corectitudinea globală a programului<sup>1)</sup></li> </ul>	<p><b>10 p.</b></p> <p>1+1 p.</p> <p>1 p.</p> <p>2 p.</p> <p>2 p.</p> <p>2 p.</p> <p>1 p.</p>	<p>O posibilă soluție se bazează pe secvența:</p> <pre> <b>pentru j←1,n execută</b>   <b>pentru i←1,n execută</b>     <b>dacă i=j atunci A[i,j]←0</b>     <b>altfel A[i,j]←n+1-j</b>   </pre> <p>(*) Se acordă numai 1 p. dacă diagonala a fost completată corect, dar ulterior elementele ei au fost suprascrise</p> <p>(**) Se acordă 1 p. dacă se respectă ordinea cerută a valorilor, și încă 1 p. pentru poziționarea acestora pe coloane.</p>

**SUBIECTUL al III-lea**

**(30 de puncte)**

<p><b>1)</b></p>	<p><b>a</b></p>	<p><b>6 p.</b></p>	<p>Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.</p>
<p><b>2)</b></p>	<p><b>Răspuns: ABACABA</b></p>	<p><b>4 p.</b></p>	<p>Se acordă numai 2 p pentru oricare dintre răspunsurile ABC, ABAC, CABA</p>
<p><b>3) a)</b></p>	<p><b>Pentru rezolvare corectă</b></p> <ul style="list-style-type: none"> <li>- antet corect</li> <li>- determinarea numărului cerut *</li> <li>- obținerea unui rezultat corect pentru valori mari ale lui n **</li> <li>-returnare rezultat</li> <li>- corectitudinea globală a subprogramului<sup>1)</sup></li> </ul>	<p><b>6 p.</b></p> <p>1 p.</p> <p>2 p.</p> <p>1 p.</p> <p>1 p.</p> <p>1 p.</p>	<p>(*) Se acordă punctajul chiar dacă metoda calculează efectiv n!</p> <p>(**)</p> <p>Valoarea returnată poate fi <b>p</b>:</p> <pre> <b>p←0</b> <b>pentru i←1,n execută</b>   <b>j←i</b>   <b>cât timp j%5=0 execută</b>     <b>p←p+1</b>     <b>j←[j/5]</b>   </pre> <p>sau poate fi:</p> <p><b>p←[n/5]+[n/25]+[n/125]+...</b></p> <p>etc.</p>

	<b>b) Pentru rezolvare corectă</b> - declarare variabile, citire date, afișare rezultat - apel util și corect al subprogramului - determinarea valorii cerute *	<b>4 p.</b> 1 p. 1 p. 2 p.	(*) Se acordă numai 1p pentru o valoare n pentru care n! are cel puțin k zerouri la sfârșit, dar nu este cea mai mică valoare.
<b>4)</b>	<b>Pentru program corect</b> - declararea corectă a tuturor variabilelor - operații cu fișiere - citirea lui n și a tuturor perechilor din fișier - calculul unei puteri a lui 2 din vecinătatea intervalului * - puterea determinată aparține intervalului - tratarea cazului 0 - afișarea tuturor valorilor în formatul cerut - corectitudinea globală a programului <sup>1)</sup>	<b>10 p.</b> 1 p. 1 p. 1 p. 2 p. 1 p. 1 p. 2 p. 1 p.	(*) puterea calculată se află în interval sau în imediata apropiere a capetelor acestuia

<sup>1)</sup> Corectitudinea globală vizează structura, sintaxa, alte greșeli neprecizate în barem



**Examenul de bacalaureat 2010**  
**PROBA E - d)**  
**Proba scrisă la INFORMATICĂ, limbajul Pascal**  
**Specializarea Matematică-informatică**

**MODEL**

- ◆ Toate subiectele (I, II și III) sunt obligatorii. Se acordă 10 puncte din oficiu.
- ◆ Timpul efectiv de lucru este de 3 ore.
- ◆ În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).

**Subiectul I** **(30 de puncte)**

Pentru itemul 1 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Care este numărul total de atribuiri efectuate la executarea secvenței de instrucțiuni alăturate? (4p.)
- ```
x:=4; y:=6;  
while y=6 do  
  y:=y+1;  
if x=y then  
  x:=x+1;
```
- a. 4                                      b. 3                                      c. 2                                      d. 5

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

**2. Se consideră algoritmul alăturat descris în pseudocod:**

S-a notat cu  $x\%y$  restul împărțirii numărului natural  $x$  la numărul natural nenul  $y$  și cu  $[z]$  partea întregă a numărului real  $z$ .

a) Scrieți numărul afișat dacă se citește valoarea  $n=12939$ . (6p.)

b) Scrieți toate valorile care pot fi citite pentru variabila  $n$  astfel încât, în urma executării algoritmului, valoarea afișată să fie **2009**. (4p.)

```
citește n (număr natural)  
nr ← 0  
p ← 1  
cât timp n ≠ 0 execută  
  c ← n%10  
  dacă c > 0 și c < 9 atunci  
    c ← c+1  
  ■  
  nr ← nr+c*p  
  p ← p*10  
  n ← [n/10]  
  ■  
scrie nr
```

c) Scrieți, în pseudocod, un algoritm echivalent cu cel dat, în care să se înlocuiască structura **cât timp...execută** cu o structură repetitivă de un alt tip. (6p.)

d) Scrieți programul **Pascal** corespunzător algoritmului dat. (10p.)

**Subiectul al II-lea**

**(30 de puncte)**

**Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.**

1. Se consideră un graf orientat cu 6 noduri numerotate de la 1 la 6 și cu mulțimea arcelor formată **doar** din arcele:
- de la fiecare nod numerotat cu un număr neprim  $i$  ( $i > 1$ ) la toate nodurile numerotate cu numere ce aparțin mulțimii divizorilor proprii ai lui  $i$  (divizori diferiți de 1 și de  $i$ )
  - de la nodul numerotat cu 1 la nodul numerotat cu 6
  - de la fiecare nod numerotat cu un număr prim  $i$  la nodul numerotat cu  $i-1$
- Pentru graful dat, câte dintre nodurile grafului au gradul exterior strict mai mare decât gradul interior? **(4p.)**
- a. 1                      b. 2                      c. 4                      d. 3
2. Câte frunze are arborele cu rădăcină descris prin următorul vector "de tați":  
**(6,5,5,2,0,3,3,8,7,7)?** **(4p.)**
- a. 1                      b. 2                      c. 5                      d. 4

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

3. În declarația alăturată, câmpurile  $x$  și  $y$  ale înregistrării pot memora numărătorul, respectiv numitorul unei fracții. Scrieți secvența de instrucțiuni prin executarea căreia se construiește în variabila  $f$  o fracție obținută prin însumarea fracțiilor memorate în variabilele  $f1$  și  $f2$ . **(6p.)**
- |  |                                   |
|--|-----------------------------------|
|  | <code>type fractie =record</code> |
|  | <code>  x,y:integer</code>        |
|  | <code>end;</code>                 |
|  | <code>var f,f1,f2:fractie;</code> |
4. În secvența de instrucțiuni de mai jos, variabila  $s$  memorează un șir de caractere format doar din litere ale alfabetului englez, iar variabila  $i$  este de tip **integer**.
- Știind că în urma executării secvenței s-a afișat succesiunea de caractere **eeleeneee** scrieți care este șirul de caractere memorat de variabila  $s$ . **(6p.)**
- |  |                                       |
|--|---------------------------------------|
|  | <code>for i:=1 to length(s) do</code> |
|  | <code>  write(s[i],'e');</code>       |
5. Scrieți un program **Pascal** care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 24$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane ale cărei elemente vor primi valori după cum urmează:
- elementele aflate pe diagonala principală a matricei vor primi valoarea **0**
  - elementele de pe prima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n**
  - elementele de pe a doua coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n-1**
  - ...
  - elementele de pe ultima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **1**
- Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).
- |  |                      |
|--|----------------------|
|  | <code>0 3 2 1</code> |
|  | <code>4 0 2 1</code> |
|  | <code>4 3 0 1</code> |
|  | <code>4 3 2 0</code> |
- Exemplu:** pentru  $n=4$  se va afișa matricea alăturată. **(10p.)**

**Subiectul al III-lea**

**(30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. În timpul procesului de generare a permutărilor mulțimii  $\{1,2,\dots,n\}$  prin metoda backtracking, în tabloul unidimensional  $x$  este plasat un element  $x_k$  ( $2 \leq k \leq n$ ). Acesta este considerat valid dacă este îndeplinită condiția: **(6p.)**
  - a.  $x_k \notin \{x_1, x_2, \dots, x_{k-1}\}$
  - b.  $x_k \neq x_{k-1}$
  - c.  $x_k \notin \{x_1, x_2, \dots, x_n\}$
  - d.  $x_k \neq x_{k-1}$  și  $x_k \neq x_{k+1}$

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Considerăm subprogramul recursiv definit alăturat. Ce se va afișa în urma apelului de mai jos?  
 $f('B')$ ; **(4p.)**

|                                                                                                                                          |  |
|------------------------------------------------------------------------------------------------------------------------------------------|--|
| <pre>procedure f(c:char);<br/>begin<br/>  if c&gt;'A' then f(pred(c));<br/>  write(c);<br/>  if c&gt;'A' then f(pred(c));<br/>end;</pre> |  |
|------------------------------------------------------------------------------------------------------------------------------------------|--|
3. a) Scrieți definiția unui subprogram, **nz**, cu un parametru întreg, **n** ( $0 < n \leq 12$ ), care returnează numărul zerourilor de la sfârșitul numărului  $n! = 1 \cdot 2 \cdot \dots \cdot n$ . **(6p.)**  
b) Scrieți programul **Pascal** care citește de la tastatură un număr natural **k** ( $0 < k \leq 2$ ) și determină, folosind apeluri ale subprogramului **nz**, cel mai mic număr natural **n** pentru care  $n!$  are cel puțin **k** zerouri la sfârșit. Numărul determinat se afișează pe ecran. **(4p.)**
4. Scrieți programul **Pascal** care citește din fișierul text **BAC.TXT** numărul întreg **n** ( $1 \leq n \leq 10000$ ) și un șir de **n** perechi de numere întregi **a b** ( $1 \leq a \leq b \leq 32000$ ), fiecare pereche fiind scrisă pe o linie nouă a fișierului, cu un spațiu între cele două numere. Programul afișează pe ecran pentru fiecare pereche **a,b** cel mai mare număr natural din intervalul închis  $[a,b]$  care este o putere a lui **2** sau numărul **0** dacă nu există nicio putere a lui **2** în intervalul respectiv. Numerele afișate pe ecran se scriu în linie, separate prin câte un spațiu. Un număr **p** este putere a lui **2** dacă există un număr natural **k** astfel încât  $p=2^k$ .  
**Exemplu:** dacă fișierul **BAC.TXT** conține numerele  
3  
2 69  
10 20  
19 25  
se va afișa: **64 16 0**. **(10p.)**

**Examenul de bacalaureat 2010**  
**PROBA E – d)**  
**Proba scrisă la INFORMATICĂ, limbajul C/C++**  
**Specializarea Matematică-informatică**

**MODEL**

- ◆ Toate subiectele (I, II și III) sunt obligatorii. Se acordă 10 puncte din oficiu.
- ◆ Timpul efectiv de lucru este de 3 ore.
- ◆ În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).

**Subiectul I**

**(30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Care este numărul total de atribuiri efectuate la executarea  
secvenței de instrucțiuni alăturate? (4p.)
- |      |      |      |      |
|------|------|------|------|
| a. 4 | b. 3 | c. 2 | d. 5 |
|------|------|------|------|
- ```
x=4; y=6;
while (y==6) y=y+1;
if (x==y) x=x+1;
```

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Se consideră algoritmul alăturat descris în pseudocod.
- S-a notat cu  $x\%y$  restul împărțirii numărului natural  $x$  la numărul natural nenul  $y$  și cu  $[z]$  partea întreagă a numărului real  $z$ .
- a) Scrieți numărul afișat dacă se citește valoarea  $n=12939$ . (6p.)
- b) Scrieți toate valorile care pot fi citite pentru variabila  $n$  astfel încât, în urma executării algoritmului, valoarea afișată să fie **2009**. (4p.)
- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura **cât timp...execută** cu o structură repetitivă de un alt tip. (6p.)
- d) Scrieți programul **C/C++** corespunzător algoritmului dat. (10p.)
- ```
citește n (număr natural)
nr ← 0
p ← 1
cât timp n ≠ 0 execută
  c ← n%10
  dacă c > 0 și c < 9 atunci
    c ← c+1
  nr ← nr+c*p
  p ← p*10
  n ← [n/10]
scrie nr
```

**Subiectul al II-lea** **(30 de puncte)**

**Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.**

1. Se consideră un graf orientat cu 6 noduri numerotate de la 1 la 6 și cu mulțimea arcelor formată **doar** din arcele:
  - de la fiecare nod numerotat cu un număr neprim  $i$  ( $i > 1$ ) la toate nodurile numerotate cu numere ce aparțin mulțimii divizorilor proprii ai lui  $i$  (divizori diferiți de 1 și de  $i$ )
  - de la nodul numerotat cu 1 la nodul numerotat cu 6
  - de la fiecare nod numerotat cu un număr prim  $i$  la nodul numerotat cu  $i-1$Pentru graful dat, câte dintre nodurile grafului au gradul exterior strict mai mare decât gradul interior? **(4p.)**  
a. 1                      b. 2                      c. 4                      d. 3
2. Câte frunze are arborele cu rădăcină descris prin următorul vector "de tați":  
**(6,5,5,2,0,3,3,3,8,7,7)?** **(4p.)**  
a. 1                      b. 2                      c. 5                      d. 4

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

3. În declarația alăturată, câmpurile  $x$  și  $y$  ale înregistrării pot memora numărătorul, respectiv numitorul unei fracții. Scrieți secvența de instrucțiuni prin executarea căreia se construiește în variabila  $f$  o fracție obținută prin însumarea fracțiilor memorate în variabilele  $f1$  și  $f2$ . **(6p.)**  

|                                                                              |                                                                |
|------------------------------------------------------------------------------|----------------------------------------------------------------|
| <pre>n=stren(s);<br/>for(i=0;i&lt;n;i++)<br/>printf("%c%c",s[i], 'e');</pre> | <pre>struct fracție<br/>{<br/>    int x,y;<br/>}f,f1,f2;</pre> |
|------------------------------------------------------------------------------|----------------------------------------------------------------|
  4. În secvența de instrucțiuni de mai jos, variabila  $s$  memorează un șir de caractere format doar din litere ale alfabetului englez, iar variabilele  $i$  și  $n$  sunt de tip **int**. Știind că în urma executării secvenței s-a afișat succesiv de caractere **eeleeneee** scrieți care este șirul de caractere memorat de variabila  $s$ . **(6p.)**
  5. Scrieți un program **C/C++** care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 24$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane ale cărei elemente vor primi valori după cum urmează:
    - elementele aflate pe diagonala principală a matricei vor primi valoarea **0**
    - elementele de pe prima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n**
    - elementele de pe a doua coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n-1**
    - ...
    - elementele de pe ultima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **1**Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).  
**Exemplu:** pentru  $n=4$  se va afișa matricea alăturată. **(10p.)**
- |         |         |
|---------|---------|
| 0 3 2 1 | 0 3 2 1 |
| 4 0 2 1 | 4 0 2 1 |
| 4 3 0 1 | 4 3 0 1 |
| 4 3 2 0 | 4 3 2 0 |

**Subiectul al III-lea**

**(30 de puncte)**

**Pentru itemul 1 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. În timpul procesului de generare a permutărilor mulțimii  $\{1,2,\dots,n\}$  prin metoda backtracking, în tabloul unidimensional  $x$  este plasat un element  $x_k$  ( $2 \leq k \leq n$ ). Acesta este considerat valid dacă este îndeplinită condiția: **(6p.)**

a.  $x_k \notin \{x_1, x_2, \dots, x_{k-1}\}$

b.  $x_k \neq x_{k-1}$

c.  $x_k \notin \{x_1, x_2, \dots, x_n\}$

d.  $x_k \neq x_{k-1}$  și  $x_k \neq x_{k+1}$

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Considerăm subprogramul recursiv definit alăturat. Ce se va afișa în urma apelului de mai jos?  
`f('B');` **(4p.)**
- ```
void f(char c)
{ if (c>'A') f(c-1);
  cout<<c; | printf("%c",c);
  if (c>'A') f(c-1);
}
```
3. a) Scrieți definiția unui subprogram, **nz**, cu un parametru întreg, **n** ( $0 < n \leq 12$ ), care returnează numărul zerourilor de la sfârșitul numărului  $n! = 1 \cdot 2 \cdot \dots \cdot n$ . **(6p.)**  
b) Scrieți programul **C/C++** care citește de la tastatură un număr natural **k** ( $0 < k \leq 2$ ) și determină, folosind apeluri ale subprogramului **nz**, cel mai mic număr natural **n** pentru care  $n!$  are cel puțin **k** zerouri la sfârșit. Numărul determinat se afișează pe ecran. **(4p.)**
4. Scrieți programul **C/C++** care citește din fișierul text **BAC.TXT** numărul întreg **n** ( $1 \leq n \leq 10000$ ) și un șir de **n** perechi de numere întregi **a b** ( $1 \leq a \leq b \leq 32000$ ), fiecare pereche fiind scrisă pe o linie nouă a fișierului, cu un spațiu între cele două numere. Programul afișează pe ecran pentru fiecare pereche **a,b** cel mai mare număr natural din intervalul închis **[a,b]** care este o putere a lui **2** sau numărul **0** dacă nu există nicio putere a lui **2** în intervalul respectiv. Numerele afișate pe ecran se scriu în linie, separate prin câte un spațiu. Un număr **p** este putere a lui **2** dacă există un număr natural **k** astfel încât  $p = 2^k$ .  
**Exemplu:** dacă fișierul **BAC.TXT** conține numerele  
**3**  
**2 69**  
**10 20**  
**19 25**  
se va afișa: **64 16 0**. **(10p.)**

**Examenul de bacalaureat 2010**  
**Proba E – d)**  
**Proba scrisă la INFORMATICĂ**  
**specializarea matematică-informatică**

**BAREM DE EVALUARE ȘI DE NOTARE**  
**(comun pentru limbajele Pascal și C/C++)**

**MODEL**

- Se punctează oricare alte formulări/ modalități de rezolvare corectă a cerințelor.
- Nu se acordă punctaje intermediare, altele decât cele precizate explicit prin barem. Nu se acordă fracțiuni de punct.
- Se acordă 10 puncte din oficiu. Nota finală se calculează prin împărțirea punctajului total acordat pentru lucrare la 10.
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.
- Utilizarea unui tip de date care depășește domeniul de valori precizat în enunț este acceptată dacă acest lucru nu afectează corectitudinea în funcționarea programului.

**SUBIECTUL I**

**(30 de puncte)**

1.	b	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
2.	a) 23949	6 p.	
	b) 1008, 1009 *	4 p.	(*) Pentru fiecare număr corect se acordă 2p
	c) Pentru program pseudocod corect - structură repetitivă corectă * - echivalența prelucrării realizate - algoritm complet - corectitudine globală	6 p. 2 p. 2 p. 1 p. 1 p.	(*) Se va puncta orice formă corectă de structură repetitivă de alt tip (de exemplu execută...cât timp, repetă...până când etc.)
	d) Pentru program corect - declararea corectă a tuturor variabilelor - citire și scriere corecte - structură repetitivă cu test inițial corectă - structură de decizie corectă - atribuiri corecte - corectitudinea globală a programului <sup>1)</sup>	10 p. 2 p. 2 p. 2 p. 2 p. 1 p. 1 p.	Elevii nu vor fi depunctați pentru modul de aliniere al instrucțiunilor.

**SUBIECTUL al II-lea**

**(30 de puncte)**

1)	b	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
2)	c	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.

<p><b>3) Pentru rezolvare corectă*</b></p> <ul style="list-style-type: none"> <li>- accesul corect la câmpurile înregistrării</li> <li>- calculul corect al număratorului **</li> <li>- calculul corect al numitorului **</li> <li>- corectitudinea sintactică a secvenței</li> </ul>	<p><b>6 p.</b></p> <p>2 p.</p> <p>2 p.</p> <p>1 p.</p> <p>1 p.</p>	<p>(*) O posibilă soluție constă în transcrierea în limbaj de programare a instrucțiunilor de mai jos:</p> <p><b>f.x←f1.x*f2.y+f1.y*f2.x</b></p> <p><b>f.y←f1.y*f2.y</b></p> <p>Se acordă punctajul maxim chiar dacă se fac prelucrări suplimentare (de exemplu simplificări, etc.) care nu denaturează rezultatul, în raport cu cerința.</p> <p>(**) Se acordă numai 1 din 3 p.dacă formulele sunt corecte, dar se confundă termenii numitor / numărator.</p>
<p><b>4) Răspuns corect: elene</b></p>	<p><b>6 p.</b></p>	<p>Dacă răspunsul diferă printr-o singură literă de răspunsul corect, se acordă numai 3 p.</p>
<p><b>5) Pentru program se acordă punctaj maxim</b></p> <ul style="list-style-type: none"> <li>- declararea corectă a tuturor variabilelor (matrice și variabile simple)</li> <li>- citire corectă</li> <li>- completarea diagonalei cu 0 *</li> <li>-completarea matricei cu valorile nenule impuse **</li> <li>- afișarea matricei în formatul cerut</li> <li>- corectitudinea globală a programului<sup>1)</sup></li> </ul>	<p><b>10 p.</b></p> <p>1+1 p.</p> <p>1 p.</p> <p>2 p.</p> <p>2 p.</p> <p>2 p.</p> <p>1 p.</p>	<p>O posibilă soluție se bazează pe secvența:</p> <pre> <b>pentru j←1,n execută</b>   <b>pentru i←1,n execută</b>     <b>dacă i=j atunci A[i,j]←0</b>     <b>altfel A[i,j]←n+1-j</b>   </pre> <p>(*) Se acordă numai 1 p. dacă diagonala a fost completată corect, dar ulterior elementele ei au fost suprascrise</p> <p>(**) Se acordă 1 p. dacă se respectă ordinea cerută a valorilor, și încă 1 p. pentru poziționarea acestora pe coloane.</p>

**SUBIECTUL al III-lea**

**(30 de puncte)**

<p><b>1)</b></p>	<p><b>a</b></p>	<p><b>6 p.</b></p>	<p>Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.</p>
<p><b>2)</b></p>	<p><b>Răspuns: ABA</b></p>	<p><b>4 p.</b></p>	<p>Se acordă numai 2 p dacă se omite o singură literă.</p>
<p><b>3) a)</b></p>	<p><b>Pentru rezolvare corectă</b></p> <ul style="list-style-type: none"> <li>- antet corect (structură+parametru)</li> <li>- determinarea numărului cerut</li> <li>- returnare rezultat</li> <li>- corectitudinea globală a subprogramului<sup>1)</sup></li> </ul>	<p><b>6 p.</b></p> <p>1+1 p.</p> <p>2 p.</p> <p>1 p.</p> <p>1 p.</p>	<p>O soluție posibilă poate fi calculul efectiv al produsului și apoi determinarea numărului de 0 de la finalul valorii obținute</p>
<p><b>b)</b></p>	<p><b>Pentru rezolvare corectă</b></p> <ul style="list-style-type: none"> <li>- declarare variabile, citire date, afișare rezultat</li> <li>- apel util și corect al subprogramului</li> <li>- determinarea valorii cerute *</li> </ul>	<p><b>4 p.</b></p> <p>1 p.</p> <p>1 p.</p> <p>2 p.</p>	<p>(*) Se acordă numai 1p pentru o valoare n pentru care n! are cel puțin k zerouri la sfârșit, dar nu este cea mai mică valoare.</p>



<b>4)</b>	<b>Pentru program corect</b>	<b>10 p.</b>	(*) puterea calculată se află în interval sau în imediata apropiere a capetelor acestuia
	- declararea corectă a tuturor variabilelor	1 p.	
	- operații cu fișiere	1 p.	
	- citirea lui $n$ și a tuturor perechilor din fișier	1 p.	
	- calculul unei puteri a lui 2 din vecinătatea intervalului *	2 p.	
	- puterea determinată aparține intervalului	1 p.	
	- tratarea cazului 0	1 p.	
	- afișarea tuturor valorilor în formatul cerut	2 p.	
- corectitudinea globală a programului <sup>1)</sup>	1 p.		

<sup>1)</sup> Corectitudinea globală vizează structura, sintaxa, alte greșeli neprecizate în barem

**Examenul de bacalaureat 2010**  
**PROBA E – d)**  
**Proba scrisă la INFORMATICĂ, limbajul Pascal**  
**Specializarea științe ale naturii**

**MODEL**

- ◆ Toate subiectele (I, II și III) sunt obligatorii. Se acordă 10 puncte din oficiu.
- ◆ Timpul efectiv de lucru este de 3 ore.
- ◆ În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).

**Subiectul I** **(30 de puncte)**

Pentru itemul 1 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Care este numărul total de atribuiri efectuate la executarea  
secvenței de instrucțiuni alăturate? (4p.)
- |                             |  |  |  |
|-----------------------------|--|--|--|
| <b>x:=4; y:=6;</b>          |  |  |  |
| <b>while y=6 do y:=y+1;</b> |  |  |  |
| <b>if x=y then x:=x+1;</b>  |  |  |  |
- a. 4                                      b. 3                                      c. 2                                      d. 5

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Se consideră algoritmul alăturat descris în pseudocod.
- S-a notat cu  $x\%y$  restul împărțirii numărului natural  $x$  la numărul natural nenul  $y$  și cu  $[z]$  partea întreagă a numărului real  $z$ .
- |  |   |
|--|---|
| a) Scrieți numărul afișat dacă se citește valoarea $n=12939$ . <span style="float: right;">(6p.)</span>  | citește $n$ (număr natural)<br>$nr \leftarrow 0$<br>$p \leftarrow 1$<br>cât timp $n \neq 0$ execută<br>$c \leftarrow n \% 10$<br>dacă $c > 0$ și $c < 9$ atunci<br>$c \leftarrow c + 1$<br>■<br>$nr \leftarrow nr + c * p$<br>$p \leftarrow p * 10$<br>$n \leftarrow [n / 10]$<br>■<br>scrie $nr$ |
| b) Scrieți toate valorile care pot fi citite pentru variabila $n$ astfel încât, în urma executării algoritmului, valoarea afișată să fie <b>2009</b> . <span style="float: right;">(4p.)</span>              |   |
| c) Scrieți, în pseudocod, un algoritm echivalent cu cel dat, în care să se înlocuiască structura <b>cât timp...execută</b> cu o structură repetitivă de un alt tip. <span style="float: right;">(6p.)</span> |   |
| d) Scrieți programul <b>Pascal</b> corespunzător algoritmului dat. <span style="float: right;">(10p.)</span>   |   |

**Subiectul al II-lea (30 de puncte)**

**Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.**

1. Care dintre apelurile de mai jos returnează rădăcina pătrată a valorii variabilei reale  $x$ ? **(4p.)**  
a  $\text{sqr}(4*x)$       b  $\text{sqrt}(x)$       c  $\text{exp}(x,4)$       d  $\text{rad}(x)$
2. Variabilele  $x1$ ,  $y1$ ,  $x2$ ,  $y2$ ,  $x$  sunt de tip întreg:  $x1$  și  $y1$  memorează numărătorul, respectiv numitorul unei fracții algebrice, iar  $x2$  și  $y2$  memorează numărătorul, respectiv numitorul unei alte fracții algebrice. Care dintre instrucțiunile de mai jos atribuie variabilei  $x$  valoarea număratorului unei fracții obținute prin însumarea celor două fracții menționate în enunț? **(4p.)**

- a.  $x:=x1*x2+y2*y1;$       b.  $x:=x1*y1+x2*y2;$
- c.  $x:=x1*y2+x2*y1;$       d.  $x:=(x1+x2)/y1/y2;$

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

3. În secvența alăturată, toate variabilele sunt de tip integer. Cu ce se poate înlocui zona punctată astfel încât, în urma executării secvenței, să se afișeze cel mai mare divizor comun al valorilor memorate inițial în variabilele  $a$  și  $b$ ? **(6p.)**  

```
while b<>0 do  
begin  
  r:=a mod b;  
  a:=b; b:=r  
end;  
.....
```
4. Se citește un număr natural  $n$  ( $0 < n \leq 12$ ) și se cere să se afișeze numărul zerourilor de la sfârșitul numărului  $n! = 1 \cdot 2 \cdot \dots \cdot n$ .  
a) Care sunt datele de intrare și care sunt datele de ieșire ale problemei enunțate? **(6p.)**  
b) Scrieți în pseudocod algoritmul de rezolvare al problemei enunțate. **(10p.)**

**Subiectul al III-lea (30 de puncte)**

**Pentru itemul 1 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Se consideră un tablou unidimensional în care elementele sunt, în ordine, (1,3,5,7,10,16,21). Pentru a afla indicele elementului din tablou cu valoarea  $x=10$ , se aplică metoda căutării binare. Care este succesiunea corectă de elemente a căror valoare se compară cu valoarea lui  $x$  pe parcursul aplicării metodei indicate? (4p.)
- a. 21,16,10      b. 7,16,10      c. 1,3,5,7,10      d. 5,7,10

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. În secvența alăturată, variabilele  $s$  și  $m$  sunt de tip **char**, iar celelalte variabile sunt de tip **integer**. Scrieți ce se afișează în urma executării secvenței, știind că de la tastatură s-au introdus următoarele 6 litere:  $x, e, r, x, e, s$ . (6p.)
- ```
m:='a'; n:=0;
for i:=1 to 6 do
begin read(s);
if m<s then begin m:=s;n:=1 end
else if m=s then n:=n+1
end;
write(m,' ',n);
```
3. Se consideră șirul lui Fibonacci, în care primii doi termeni sunt  $f_1=1, f_2=1$ , iar al  $k$ -lea ( $k>2$ ) termen se calculează cu ajutorul formulei  $f_k=f_{k-1}+f_{k-2}$ . Care este valoarea celui de-al șaptelea termen al șirului,  $f_7$ ? (4p.)
4. Scrieți un program **Pascal** care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 24$ ) și construiește în memorie un tablou unidimensional cu  $n$  elemente, numerotate de la 1 la  $n$ , care vor primi valori după cum urmează:  
- elementele aflate pe poziții pare vor primi valoarea 0;  
- elementele aflate pe poziții impare vor primi o valoare egală cu poziția pe care o ocupă în tablou.  
Programul va afișa tabloul astfel construit pe ecran, cu câte un spațiu între elementele alăturate.  
**Exemplu:** pentru  $n=5$  se va afișa tabloul: 1 0 3 0 5 (6p.)
5. Fișierul text **BAC.TXT** conține pe prima linie numărul întreg  $n$  ( $1 \leq n \leq 10000$ ) și pe a doua linie un șir de  $n$  numere întregi.  
Scrieți programul **Pascal** care citește numerele din fișier și afișează pe ecran numai numerele de pe a doua linie a fișierului care sunt puteri ale lui 2 sau numărul 0 dacă nu există nicio putere a lui 2 printre acestea. Numerele afișate pe ecran se scriu în linie, separate prin câte un spațiu. Un număr  $p$  este putere a lui 2 dacă există un număr natural  $k$  astfel încât  $p=2^k$ . (10p.)

**Exemplu:** dacă fișierul **BAC.TXT** conține numerele alăturate, se va afișa: 2 64 16

```
7
2 64 10 20 16 25 49
```

**Examenul de bacalaureat 2010**  
**PROBA E – d)**  
**Proba scrisă la INFORMATICĂ, limbajul C/C++**  
**Specializarea științe ale naturii**

**MODEL**

- ◆ Toate subiectele (I, II și III) sunt obligatorii. Se acordă 10 puncte din oficiu.
- ◆ Timpul efectiv de lucru este de 3 ore.
- ◆ În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).

**Subiectul I** **(30 de puncte)**

Pentru itemul 1 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Care este numărul total de atribuiri efectuate la executarea  
secvenței de instrucțiuni alăturate? (4p.)
- ```
x=4; y=6;  
while (y==6) y=y+1;  
if (x==y) x=x+1;
```
- a. 4                                      b. 3                                      c. 2                                      d. 5

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Se consideră algoritmul alăturat descris în pseudocod.
- S-a notat cu  $x\%y$  restul împărțirii numărului natural  $x$  la numărul natural nenul  $y$  și cu  $[z]$  partea întreagă a numărului real  $z$ .
- a) Scrieți numărul afișat dacă se citește valoarea  $n=12939$ . (6p.)
- b) Scrieți toate valorile care pot fi citite pentru variabila  $n$  astfel încât, în urma executării algoritmului, valoarea afișată să fie **2009**. (4p.)
- c) Scrieți, în pseudocod, un algoritm echivalent cu cel dat, în care să se înlocuiască structura **cât timp...execută** cu o structură repetitivă de un alt tip. (6p.)
- d) Scrieți programul **C/C++** corespunzător algoritmului dat. (10p.)
- ```
citește n (număr natural)  
nr ← 0  
p ← 1  
cât timp n ≠ 0 execută  
  c ← n%10  
  dacă c > 0 și c < 9 atunci  
    c ← c+1  
  nr ← nr+c*p  
  p ← p*10  
  n ← [n/10]  
scrie nr
```

**Subiectul al II-lea (30 de puncte)**

**Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.**

1. Care dintre apelurile de mai jos returnează rădăcina pătrată a valorii variabilei reale  $x$ ? (4p.)  
a  $\text{sqr}(4*x)$       b  $\text{sqrt}(x)$       c  $\text{pow}(x,4)$       d  $\text{rad}(x)$
2. Variabilele  $x_1$ ,  $y_1$ ,  $x_2$ ,  $y_2$ ,  $x$  sunt de tip întreg:  $x_1$  și  $y_1$  memorează numărătorul, respectiv numitorul unei fracții algebrice, iar  $x_2$  și  $y_2$  memorează numărătorul, respectiv numitorul unei alte fracții algebrice. Care dintre instrucțiunile de mai jos atribuie variabilei  $x$  valoarea număratorului unei fracții obținute prin însumarea celor două fracții menționate în enunț? (4p.)

- a.  $x=x_1*x_2+y_2*y_1$ ;
- b.  $x=x_1*y_1+x_2*y_2$ ;
- c.  $x=x_1*y_2+x_2*y_1$ ;
- d.  $x=(x_1+x_2)/y_1/y_2$ ;

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

3. În secvența alăturată, toate variabilele sunt de tip int. Cu ce  
se poate înlocui zona punctată astfel încât, în urma executării  
secvenței, să se afișeze cel mai mare divizor comun al  
valorilor memorate în variabilele a și b? (6p.)  

```
while (b!=0)
{ r=a%b; a=b; b=r; }
.....
```
4. Se citește un număr natural  $n$  ( $0 < n \leq 12$ ) și se cere să se afișeze numărul zerourilor de la sfârșitul numărului  $n! = 1 \cdot 2 \cdot \dots \cdot n$ .  
a) Care sunt datele de intrare și care sunt datele de ieșire ale problemei enunțate? (6p.)  
b) Scrieți în pseudocod algoritmul de rezolvare al problemei enunțate. (10p.)

**Subiectul al III-lea (30 de puncte)**

**Pentru itemul 1 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Se consideră un tablou unidimensional în care elementele sunt, în ordine, (1,3,5,7,10,16,21). Pentru a afla indicele elementului din tablou cu valoarea  $x=10$ , se aplică metoda căutării binare. Care este succesiunea corectă de elemente a căror valoare se compară cu valoarea lui  $x$  pe parcursul aplicării metodei indicate? (4p.)
- a. 21,16,10      b. 7,16,10      c. 1,3,5,7,10      d. 5,7,10

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. În secvența de instrucțiuni  
alăturată, variabilele  $s$  și  $m$  sunt de tip `char`, iar celelalte variabile sunt de tip `int`.
- ```
m='a'; n=0;
for(i=1;i<=6;i++)
{ scanf("%c",&s); | cin>>s;
  if(m<s) {m=s;n=1;}
  else if(m==s) n++;
}
printf("%c %d",m,n); | cout<<m<<' '<<n;
```
- Scrieți ce se afișează în urma executării secvenței, știind că de la tastatură s-au introdus următoarele 6 litere:  $x, e, r, x, e, s$ . (6p.)
3. Se consideră șirul lui Fibonacci, în care primii doi termeni sunt  $f_1=1, f_2=1$ , iar al  $k$ -lea ( $k>2$ ) termen se calculează cu ajutorul formulei  $f_k=f_{k-1}+f_{k-2}$ . Care este valoarea celui de-al șaptelea termen al șirului,  $f_7$ ? (4p.)
4. Scrieți un program `C/C++` care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 24$ ) și construiește în memorie un tablou unidimensional cu  $n$  elemente, numerotate de la 1 la  $n$ , care vor primi valori după cum urmează:  
- elementele aflate pe poziții pare vor primi valoarea 0;  
- elementele aflate pe poziții impare vor primi o valoare egală cu poziția pe care o ocupă în tablou.  
Programul va afișa tabloul astfel construit pe ecran, cu câte un spațiu între elementele alăturate.  
**Exemplu:** pentru  $n=5$  se va afișa tabloul: 1 0 3 0 5 (6p.)
5. Fișierul text `BAC.TXT` conține pe prima linie numărul întreg  $n$  ( $1 \leq n \leq 10000$ ) și pe a doua linie un șir de  $n$  numere întregi.  
Scrieți programul `C/C++` care citește numerele din fișier și afișează pe ecran numai numerele de pe a doua linie a fișierului care sunt puteri ale lui 2 sau numărul 0 dacă nu există nicio putere a lui 2 printre acestea. Numerele afișate pe ecran se scriu în linie, separate prin câte un spațiu. Un număr  $p$  este putere a lui 2 dacă există un număr natural  $k$  astfel încât  $p=2^k$ . (10p.)

**Exemplu:** dacă fișierul `BAC.TXT` conține  
numerele alăturate, se va afișa: 2 64 16      | 7  
2 64 10 20 16 25 49

**Examenul de bacalaureat 2010**  
**Proba E – d)**  
**Proba scrisă la INFORMATICĂ**  
**specializarea științe ale naturii**

**BAREM DE EVALUARE ȘI DE NOTARE**  
**(comun pentru limbajele Pascal și C/C++)**

**MODEL**

- Se punctează oricare alte formulări/ modalități de rezolvare corectă a cerințelor.
- Nu se acordă punctaje intermediare, altele decât cele precizate explicit prin barem. Nu se acordă fracțiuni de punct.
- Se acordă 10 puncte din oficiu. Nota finală se calculează prin împărțirea punctajului total acordat pentru lucrare la 10.
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.
- Utilizarea unui tip de date care depășește domeniul de valori precizat în enunț este acceptată dacă acest lucru nu afectează corectitudinea în funcționarea programului.

**SUBIECTUL I**

**(30 de puncte)**

1.	b	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
2.	a) 23949	6 p.	
	b) 1008, 1009 *	4 p.	(*) Pentru fiecare număr corect se acordă 2p
	c) Pentru program pseudocod corect - structură repetitivă corectă * - echivalența prelucrării realizate - algoritm complet - corectitudine globală	6 p. 2 p. 2 p. 1 p. 1 p.	(*) Se va puncta orice formă corectă de structură repetitivă de alt tip (de exemplu execută...cât timp, repetă...până când etc.)
	d) Pentru program corect -declarația corectă a tuturor variabilelor -citire și scriere corecte -structură repetitivă cu test inițial corectă -structură de decizie corectă -atribuiri corecte -corectitudinea globală a programului <sup>1)</sup>	10 p. 2 p. 2 p. 2 p. 2 p. 1 p. 1 p.	Elevii nu vor fi depunctați pentru modul de aliniere al instrucțiunilor.

**SUBIECTUL al II-lea**

**(30 de puncte)**

1)	b	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
2)	c	4 p.	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
3)	Pentru răspuns corect - pentru instrucțiune corectă de scriere -pentru utilizarea variabilei corecte	6 p. 3 p. 3 p.	O soluție posibilă poate fi transcrierea în limbaj de programare a comenzii <b>scrie a</b>



4)	a) <b>Pentru răspuns corect</b> - date de intrare identificate corect - date de ieșire identificate corect	<b>6 p.</b> 3 p. 3 p.	
	b) <b>Pentru rezolvare corectă</b> - citirea datelor - determinarea numărului cerut - scrierea corectă a structurilor de control - scrierea rezultatului	<b>10 p.</b> 1 p. 5 p. 3 p. 1 p.	O soluție posibilă poate fi calculul efectiv al produsului și apoi determinarea numărului de 0 de la finalul valorii obținute (* Se va puncta orice formă corectă de structură repetitivă (de exemplu execută...cât timp, execută... până când, repetă...până când etc.) sau decizională

**SUBIECTUL al III-lea**

**(30 de puncte)**

1)	<b>b</b>	<b>4 p.</b>	Pentru orice alt răspuns (chiar dacă acesta include și litera corectă) se acordă 0 puncte.
2)	<b>Răspuns: x 2</b>	<b>6 p.</b>	Se acordă 3 p pentru litera corectă și 3 p pentru numărul corect de apariții ale acesteia.
3)	<b>18</b>	<b>4 p.</b>	
4)	<b>Pentru rezolvare corectă</b> - declarare corectă a variabilelor - completarea corectă a elementelor de pe poziții pare - completarea corectă a elementelor de pe poziții impare - completarea tuturor elementelor (*) - afișarea corectă a tuturor elementelor în formatul cerut - corectitudinea globală a programului <sup>1)</sup>	<b>6 p.</b> 1 p. 1 p. 1 p. 1 p. 1 p. 1 p.	(* se va acorda punctajul chiar dacă elementele nu sunt completate cu valorile cerute
5) a)	<b>Pentru program corect</b> - declararea corectă a tuturor variabilelor - operații cu fișiere - citirea lui n și a tuturor numerelor - verificarea proprietății de putere a lui 2 - tratarea cazului 0 - afișarea tuturor valorilor în formatul cerut - corectitudinea globală a programului <sup>1)</sup>	<b>10 p.</b> 1 p. 1 p. 2 p. 2 p. 1 p. 2 p. 1 p.	(* puterea calculată se află în interval sau în imediata apropiere a capetelor acestuia

<sup>1)</sup> Corectitudinea globală vizează structura, sintaxa, alte greșeli neprecizate în barem