

Examenul de bacalaureat național 2015
Proba E. d)
Informatică
Limbajul C/C++

Varianta 5

*Filiera teoretică, profilul real, specializările: matematică-informatică
matematică-informatică intensiv informatică*
Filiera vocațională, profilul militar, specializarea matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv 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).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.

SUBIECTUL I **(30 de puncte)**

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

1. Expresia C/C++ alăturată are valoarea: **(4p.)** | $4+5\%7*2$
a. 4 b. 8 c. 9 d. 14

2. Se consideră algoritmul alăturat, reprezentat în pseudocod.

S-a notat cu $a\%b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .

- a) Scrieți valoarea afișată dacă se citesc, în această ordine, numerele 997 și 1005. **(6p.)**
- b) Dacă pentru m se citește numărul 54321, scrieți cel mai mare număr care poate fi citit pentru n astfel încât, în urma executării algoritmului, valoarea afișată să fie 0. **(4p.)**
- c) Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind structura **pentru...execută** cu o structură de tip **cât timp...execută**. **(6p.)**
- d) Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**

```
citește m,n
(numere naturale,  $m \leq n$ )
nr ← 0
pentru x ← m, n execută
| y ← 0
| z ← x
| repetă
| | y ← y*10+z%10
| | z ← [z/10]
| până când z=0
| dacă x=y atunci
| | nr ← nr+1
| ■
| ■
scrie nr
```

SUBIECTUL al II-lea

(30 de puncte)

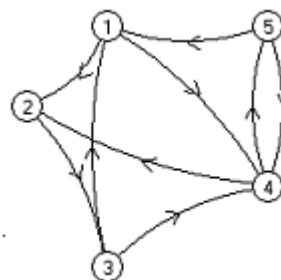
Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. În declarațiile alăturate, variabilele **A** și **B** memorează coordonatele câte unui punct în sistemul de coordonate **xOy**. Indicați expresia care are valoarea 1 dacă și numai dacă cele două puncte coincid. **(4p.)**

```
struct punct  
{ int x,y;  
}A,B;
```

- a. **A[x]==B[x] && A[y]==B[y]**
- b. **A.x==B.x && A.y==B.y**
- c. **x[A]==x[B] && y[A]==y[B]**
- d. **x.A==x.B && y.A==y.B**

2. Un graf orientat cu 5 vârfuri, numerotate de la 1 la 5, este reprezentat alăturat. Numărul maxim de arce care se pot elimina, astfel încât graful parțial obținut să fie tare conex este: **(4p.)**



- a. 2
- b. 3
- c. 4
- d. 5

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

3. Un arbore cu 8 noduri, numerotate de la 1 la 8, este reprezentat prin vectorul de „tați” (4, 4, 0, 3, 6, 3, 2, 7). Scrieți un lanț elementar care are o extremitate în rădăcina arborelui și cealaltă extremitate în una dintre frunzele acestuia. **(6p.)**

4. Variabilele **i** și **j** sunt de tip întreg, iar variabila **a** memorează un tablou bidimensional cu 5 linii și 5 coloane, numerotate de la 1 la 5, având inițial toate elementele nule.

```
Fără a utiliza alte variabile decât cele menționate, scrieți secvența de  
instrucțiuni de mai jos, înlocuind punctele de suspensie astfel încât, în  
urma executării secvenței obținute, variabila a să memoreze tabloul  
alăturat.  
for(i=1;i<=5;i++)  
  for(j=1;j<=5;j++)  
    .....
```

1	0	1	0	1
0	1	0	1	0
1	0	1	0	1
0	1	0	1	0
1	0	1	0	1

(6p.)

5. Se consideră un text cu cel mult 100 de caractere, în care cuvintele sunt formate numai din litere mari și mici ale alfabetului englez și sunt separate prin câte un spațiu. Textul reprezintă numele unei instituții sau al unei organizații.

Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și construiește în memorie, apoi afișează pe ecran, un șir de caractere ce reprezintă acronimul corespunzător numelui citit. Acronimul este format din primul caracter al fiecărui cuvânt al numelui care începe cu majusculă.

Exemplu: dacă șirul citit este
Universitatea de Arte Plastice BUCURESTI
se va obține șirul
UAPB

(10p.)

SUBIECTUL al III-lea

(30 de puncte)

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

1. Utilizând metoda backtracking, se generează toate șiragurile formate din câte 5 pietre distincte din mulțimea {opal, rubin, safir, smarald, topaz}. Două șiraguri sunt distincte dacă pietrele sunt dispuse în altă ordine. Primele patru soluții obținute sunt, în această ordine: (opal, rubin, safir, smarald, topaz), (opal, rubin, safir, topaz, smarald), (opal, rubin, smarald, safir, topaz) și (opal, rubin, smarald, topaz, safir). Indicați soluția care trebuie eliminată din enumerarea următoare, astfel încât cele rămase să apară în ordinea generării lor, pe poziții consecutive: (smarald, safir, opal, topaz, rubin) (smarald, safir, topaz, opal, rubin) (smarald, safir, topaz, rubin, opal) (smarald, topaz, opal, rubin, safir). **(4p.)**
- a. (smarald, safir, opal, topaz, rubin) b. (smarald, safir, topaz, opal, rubin)
c. (smarald, safir, topaz, rubin, opal) d. (smarald, topaz, opal, rubin, safir)

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

2. Subprogramul **F** este definit alăturat. Scrieți ce se afișează în urma apelului de mai jos. **(6p.)**
- ```
void F (long a, int b)
{ if(a*b!=0)
 if(a%2==0)
 { cout<<a%10; | printf("%d",a%10);
 F(a/10,b-1);
 }
 else
 { F(a/10,b+1);
 cout<<a%10; | printf("%d",a%10);
 }
}
```
- F(154678,3);**
3. Subprogramul **DivImpar** are doi parametri, **a** și **b**, prin care primește două numere naturale din intervalul  $[1,10^5]$ . Subprogramul returnează cel mai mare divizor comun impar al numerelor **a** și **b**. Scrieți definiția completă a subprogramului. **Exemplu:** dacă **a=30** și **b=60**, subprogramul returnează valoarea 15. **(10p.)**
4. Fișierul **date.in** conține un șir de cel mult un milion de numere naturale din intervalul  $[0,10^9]$ , separate prin câte un spațiu. Șirul are cel puțin doi termeni pari și cel puțin doi termeni impari. Se cere să se afișeze pe ecran mesajul **DA** dacă șirul aflat în fișier are un subșir ordonat crescător, format din toți termenii pari ai săi, și un subșir ordonat descrescător, format din toți termenii impari ai săi. Dacă nu există două astfel de subșiruri, programul afișează pe ecran mesajul **NU**. Pentru verificarea proprietății cerute utilizați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei necesare. **Exemplu:** dacă fișierul **date.in** conține numerele 7 2 5 2 4 3 8 se afișează pe ecran mesajul **DA** iar dacă fișierul conține numerele 5 2 7 2 4 3 8 se afișează pe ecran mesajul **NU**
- a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**  
b) Scrieți programul C/C++ corespunzător algoritmului descris. **(6p.)**