Examenul național de bacalaureat 2011
Proba E. d)
Proba scrisă la INFORMATICĂ

Limbajul C/C++
Specializările matematică informatică şi matematică informatică, intensiv informatică

MODEL

- **Toate subiectele 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. Indicați care dintre expresiile `C/C++` de mai jos are valoarea `1` dacă şi numai dacă numărul natural memorat în variabila întreagă `n` este divizibil cu `2` şi cu `3`. **(4p.)**

a. `(n/2==0) || (n/3!=0)`
b. `(n%3==2) || (n%2==3)`
c. `(n%2==0) && (n%3!=1)`
d. `(n%2!=1) && (n%3==0)`

2. Se consideră algoritmul alăturat descris în pseudocod:

S-a notat cu `x%y` restul împărțirii numărului natural **x** la numărul natural nenul `y` şi cu `[z]` partea întreagă a numărului real `z`.

```
citeşte n (număr natural nenul)
m←0
┌repetă
| c←n%10
| n←[n/10]
|┌dacă c<5 atunci
|| c←2*c
|└■
| m←m*10+c
└până când n=0
scrie m
```

a) Scrieți numărul afişat dacă pentru variabila **n** se citeşte valoarea `6451`. **(6p.)**

b) Scrieți toate numerele de cel mult trei cifre care pot fi citite pentru variabila `n` astfel încât, pentru fiecare dintre acestea, numărul afişat în urma executării algoritmului să fie `26`. **(4p.)**

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

d) Scrieți programul `C/C++` corespunzător algoritmului dat. **(10p.)**

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

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

1. Variabilele **i** şi **j** sunt de tip **int** (**1≤i≤5**, **1≤j≤5**), iar variabila **A** memorează elementele unui tablou bidimensional, cu **5** linii şi **5** coloane, numerotate de la **1** la **5**. Expresia care are valoarea **1** dacă şi numai dacă elementul **A[i][j]** se află pe diagonala secundară a tabloului este: **(4p.)**

a. `i+j==4` **b.** `i+j==6` **c.** `i<j` **d.** `i==j`

2. Numărul minim de muchii ale unui graf neorientat, conex, cu **10** de noduri, este: **(4p.)**

a. `5` **b.** `9` **c.** `10` **d.** `45`

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

3. Se consideră graful orientat, cu **5** vârfuri, numerotate de la **1** la **5**, reprezentat prin matricea de adiacență alăturată. Enumerați vârfurile grafului care au gradul extern egal cu cel intern. **(6p.)**

```
0 1 0 1 0
0 0 0 0 1
0 1 0 0 0
0 1 1 0 0
0 1 0 1 0
```

4. Se consideră tipul **elev**, definit alăturat, ce permite declararea unor variabile care să memoreze anumite date despre un elev: numele şi media sa generală anuală.

```
typedef struct {
          char nume[20];
          float medie;
        } elev;
int i;
```

Utilizând tipul **elev**, declarați o variabilă, cu numele **clasa**, care să poată memora datele menționate mai sus pentru fiecare dintre cei **30** de elevi ai unei clase, apoi scrieți instrucțiunile care pot completa zona punctată din secvența de instrucțiuni de mai jos, pentru a inițializa, prin citire de la tastatură, variabila respectivă.

```
for(i=0;i<30;i++)
  .......
```

(6p.)

5. Scrieți un program **C/C++** care citeşte de la tastatură un număr natural nenul, **n** (**n≤10**), şi **n** cuvinte, fiecare cuvânt fiind format din cel puțin **3** şi cel mult **20** de litere ale alfabetului englez. La introducerea datelor, după fiecare cuvânt se tastează Enter. Programul construieşte în memorie, şi apoi afişează pe ecran, şirul de caractere obținut prin preluarea primelor trei litere ale fiecărui cuvânt citit, în ordinea citirii acestora.

Exemplu: pentru **n=3** şi cuvintele

frumos
BUN
Intelept

se construieşte şirul de caractere **fruBUNInt** **(10p.)**

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

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

1. Se consideră subprogramele recursive **R1** şi **R2**, definite mai jos.

```
long R1(int x, int p){
  if(p==0) return 1;
  return x*R1(x,p-1);
}
```

```
long R2(int x, int p){
  long f;
  if(p==0) return 1;
  if(p%2==0) {  f=R2(x,p/2);
                return f*f;
             }
  return x*R2(x,p-1);
}
```

La apel, pentru parametrii **x=3** şi **p=5**, returnează valoarea expresiei 3^5: **(4p.)**

a. atât **R1**, cât şi **R2** **b.** numai **R1** **c.** numai **R2** **d.** nici **R1**, nici **R2**

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

2. Având la dispoziție cinci flori diferite, **lalea**, **narcisă**, **mac**, **frezie**, **garoafă**, se utilizează metoda backtracking pentru a obține toate posibilitățile de a forma buchete de câte trei flori, ştiind că în cadrul unui buchet nu contează ordinea de aşezare a florilor. Primele patru soluții obținute sunt, în această ordine: (**lalea**, **narcisă**, **mac**), (**lalea**, **narcisă**, **frezie**), (**lalea**, **narcisă**, **garoafă**), (**lalea**, **mac**, **frezie**). Scrieți ultimele două soluții generate, în ordinea obținerii lor. **(6p.)**

3. Subprogramul **eval** are trei parametri:

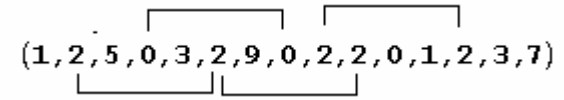
- **a** şi **b**, prin care primeşte câte un număr întreg cu cel mult patru cifre;
- **rez**, prin care furnizează numărul obținut prin însumarea valorilor transmise prin parametrii **a** şi **b**.

Exemplu: dacă **a=3**, **b=5**, în urma apelului, valoarea furnizată prin parametrul **rez** este **8**.

a) Scrieți numai antetul subprogramului **eval**. **(4p.)**

b) Fişierul **BAC.IN** conține un şir de numere întregi, cu cel mult două cifre fiecare. În fişier se află cel puțin două şi cel mult **100** de numere, separate prin câte un spațiu.
Scrieți un program **C/C++** care citeşte toate numerele din fişierul **BAC.IN** şi, utilizând apeluri utile ale subprogramului **eval**, calculează suma acestora şi apoi afişează pe ecran valoarea obținută.
Exemplu: dacă fişierul conține numerele de mai jos, se va afişa pe ecran valoarea **11**.
2 12 -7 4 **(6p.)**

4. Numim „**k**-pereche” într-un tablou unidimensional două elemente cu valori egale ale acestuia, între care se află alte **k** elemente.
Scrieți un program **C/C++** care citeşte de la tastatură două numere naturale, **n** şi **k** (**3≤n≤100**, **1≤k≤n-2**), cele **n** elemente ale unui tablou unidimensional, numere naturale, fiecare cu cel mult trei cifre, apoi afişează pe ecran numărul de „**k**-perechi” din tabloul citit.
Exemplu: dacă **n=15**, **k=3**, iar tabloul este



se va afişa pe ecran valoarea **4**. **(10p.)**