

Examenul de bacalaureat național 2020
Proba E. d)
Informatică
Limbajul C/C++

Testul 19

Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică
Filieră vocațională, profil militar, specializare matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- Identificatorii utilizați în rezolvări 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). Datele de intrare se consideră corecte, validarea lor nefiind necesară.
- În grafurile din cerințe oricare arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

SUBIECTUL I (20 de puncte)

Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

- Variabile x , y și z sunt de tip întreg. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă x , y și z au valori identice.
 - `!(x!=y) || !(y!=z)`
 - `!(x!=y!=z)`
 - `!(x!=y) || (y!=z)`
 - `!(x!=y || y!=z)`
- Subprogramul `f` este definit alăturat. Indicați de câte ori se execută subprogramul pentru apelul `f(10, 20)`.

```
int f(int x, int y)
{
    if(x<=1 || y<=1) return 0;
    if(x>y) return 1+f(f(x-y,y), y-1);
    return 1+f(x-1, f(x,y-x));
}
```

 - de 2 ori
 - de 5 ori
 - de 10 ori
 - de 20 de ori
- Utilizând metoda backtracking, se generează toate modalitățile de a pregăti o lucrare pentru un proiect, utilizând, într-o anumită ordine, toate sursele de documentare din mulțimea {**culegeri**, **manuale**, **notițe**} pentru o primă informare, apoi una dintre sursele de documentare din mulțimea {**web**, **interviuri**} pentru dezvoltarea temei, și, la final, una dintre resursele din mulțimea {**image**, **sunet**, **text**, **video**} pentru realizarea tehnică a lucrării, având în vedere următoarele restricții: imaginea și textul se pot folosi numai împreună cu site-urile web, iar sunetul și video numai împreună cu interviurile. Primele cinci soluții generate sunt, în această ordine: (**culegeri**, **manuale**, **notițe**, **web**, **image**), (**culegeri**, **manuale**, **notițe**, **web**, **text**), (**culegeri**, **manuale**, **notițe**, **interviuri**, **sunet**), (**culegeri**, **manuale**, **notițe**, **interviuri**, **video**), (**culegeri**, **notițe**, **manuale**, **web**, **image**). Indicați a șaptea soluție generată.
 - (**culegeri**, **notițe**, **manuale**, **interviuri**, **sunet**)
 - (**culegeri**, **notițe**, **manuale**, **interviuri**, **video**)
 - (**culegeri**, **notițe**, **manuale**, **web**, **text**)
 - (**manuale**, **culegeri**, **notițe**, **web**, **image**)
- Un arbore cu rădăcină, cu 8 noduri, numerotate de la 1 la 8, este reprezentat prin vectorul de „tați” (2, 5, 6, 2, 0, 5, 3, 1). Indicați un lanț cu o extremitate în rădăcină, iar cealaltă extremitate în nodul 1.
 - 6, 5, 2, 1
 - 5, 3, 1
 - 5, 2, 1
 - 3, 1
- Un graf neorientat cu 7 noduri, numerotate de la 1 la 7, are muchiile [1, 3], [1, 6], [1, 7], [2, 4], [3, 6], [3, 7], [5, 6], [6, 7]. Indicați un set format dintr-un număr minim de muchii care i se pot adăuga grafului, astfel încât graful obținut să fie hamiltonian.
 - [4, 5]
 - [1, 2]
 - [1, 2], [4, 5]
 - [1, 2], [2, 3], [4, 5]

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.
S-a notat cu $a \div b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .
- a. Scrieți ce se afișează în urma executării algoritmului dacă se citește numărul 60534. (6p.)
- b. Scrieți trei numere din intervalul $[0, 10^2]$ care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze 88. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat a doua structură `cât timp...execută` cu o structură repetitivă de alt tip. (6p.)
2. Variabila p memorează date despre un proiect: tema (un șir de maximum 20 de caractere), numărul surselor de documentare (maximum 10) și, pentru fiecare astfel de sursă, denumirea/adresa web (un șir de maximum 100 de caractere). Știind că expresiile C/C++ de mai jos au ca valori tema unui proiect, numărul de surse de documentare ale acestuia și denumirea/adresa web a primei astfel de surse, scrieți definiția unei structuri cu eticheta `proiect`, care permite memorarea datelor precizate pentru un proiect, și declarați corespunzător variabila p . (6p.)
- `p.tema p.nrSurse p.denAdrSursa[0]`
3. Variabilele i și j sunt de tip întreg, iar variabila s poate memora un șir de cel mult 20 de caractere. Scrieți șirul memorat de variabila s în urma executării secvenței de mai jos. (6p.)
- ```
strcpy(s, "informatie"); n=strlen(s)-1;
for(i=0; i<n/2; i++)
 if(strchr("aeiou", s[i])!=NULL && strchr("aeiou", s[n-i])!=NULL)
 { s[i]=s[i+1]; s[n-i]=s[n-i-1]; }
```

```
citește n (număr natural nenul)
cât timp n≠0 execută
 c←n%10; n←[n/10]
 dacă c%2≠0 atunci
 c←c+1
 ■
cât timp c>0 și c<10 execută
 scrie c
 c←c*2
 ■
 ■
```

**SUBIECTUL al III-lea**

**(30 de puncte)**

1. Subprogramul `paritate` are doi parametri:
- $n$ , prin care primește un număr natural ( $n \in [1, 10^9]$ );
  - $nr$  prin care furnizează numărul de divizori naturali ai lui  $n$  cu aceeași paritate ca  $n$ .
- Scrieți definiția completă a subprogramului. (10p.)
- Exemplu:** dacă  $n=20$ , după apel  $nr=4$  (divizorii lui 20 sunt 1, 2, 4, 5, 10, 20).
2. Într-un tablou bidimensional, cu elemente având valori numai în mulțimea  $\{0, 1\}$ , numim linii „complementare” două linii cu proprietatea că oricare două elemente ale acestora, aflate pe aceeași coloană, sunt diferite. Scrieți un program C/C++ care citește de la tastatură trei numere naturale  $m, n$  și  $k$  ( $m, n \in [2, 20], k \in [1, m]$ ) și elementele unui tablou bidimensional cu  $m$  linii și  $n$  coloane, numere naturale din mulțimea  $\{0, 1\}$ . Programul afișează pe ecran mesajul **DA** dacă există cel puțin o linie „complementară” cu linia a  $k$ -a a acestuia, sau mesajul **NU** în caz contrar. (10p.)
- Exemplu:** pentru  $m=7, n=3, k=2$  și tabloul alăturat se afișează pe ecran mesajul **DA**.
- |   |   |   |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 1 | 1 |
| 0 | 0 | 1 |
3. Fișierul `bac.txt` conține, în ordine crescătoare, cel mult  $10^6$  numere naturale din intervalul  $[0, 10^9]$ , separate prin câte un spațiu. Se cere să se afișeze pe ecran, în ordine strict crescătoare, separate prin câte un spațiu, numerele pare distincte care apar în fișier. Dacă nu există niciun astfel de număr, se afișează pe ecran mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare. (2p.)
- Exemplu:** dacă fișierul conține numerele 5 5 6 7 8 8 8 8 10 15 16 16 25 25 49 pe ecran se afișează, în această ordine, numerele 6 8 10 16 (8p.)
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
- b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)