

Examenul național de bacalaureat 2021
Proba E. d)
INFORMATICĂ
Limbajul C/C++

Testul 2

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

- Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.
- Timpul de lucru efectiv este de trei 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 muchie are extremități distincte și oricare două 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.

- Variabilele x și y sunt de tip întreg și memorează numere naturale nenule. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul memorat în x are cifra zecilor egală cu 2 și este un multiplu al numărului memorat în y .
 - $x/10\%10==2 \ \&\& \ x\%y==0$
 - $x/10\%10==2 \ || \ y\%x==0$
 - $x/2\%10==10 \ \&\& \ x\%y==0$
 - $x/2\%10==10 \ || \ y\%x==0$
- Subprogramul f este definit alăturat. Indicați ce se afișează în urma apelului de mai jos.
 $f(4)$;

```
void f(int x)
{ while(x>1){ x=x-1; f(x-1);
  cout<<x; | printf("%d",x);
}
```

 - 11111
 - 00000
 - 01010
 - 01101
- Utilizând metoda backtracking se generează toate posibilitățile de a forma selecții de câte 4 piese ale soliștilor/formațiilor din mulțimea {"Bono", "Bruce Springsteen", "Bon Jovi", "Leonard Cohen", "Portugal.The Man"}, astfel încât pe oricare două poziții alăturate să nu se afle două piese ale soliștilor/formațiilor din submulțimea {"Bono", "Bon Jovi", "Portugal.The Man"}. Primele cinci selecții generate sunt, în această ordine, ("Bono", "Bruce Springsteen", "Bono", "Bruce Springsteen"), ("Bono", "Bruce Springsteen", "Bono", "Leonard Cohen"), ("Bono", "Bruce Springsteen", "Bruce Springsteen", "Bono"), ("Bono", "Bruce Springsteen", "Bruce Springsteen", "Bruce Springsteen"), ("Bono", "Bruce Springsteen", "Bruce Springsteen", "Bon Jovi"). Ultima selecție generată este:
 - ("Portugal.The Man", "Leonard Cohen", "Portugal.The Man", "Portugal.The Man")
 - ("Portugal.The Man", "Leonard Cohen", "Portugal.The Man", "Bruce Springsteen")
 - ("Portugal.The Man", "Leonard Cohen", "Portugal.The Man", "Leonard Cohen")
 - ("Portugal.The Man", "Leonard Cohen", "Leonard Cohen", "Portugal.The Man")
- Un arbore cu 10 noduri, numerotate de la 1 la 10, este reprezentat prin vectorul de „tați” $(0, 1, 1, 1, 2, 3, 3, 7, 4, 6)$. Indicați numărul de frunze ale arborelui.
 - 3
 - 4
 - 5
 - 6
- Un graf neorientat are 10 noduri, numerotate de la 1 la 10, și muchiile $[1, 4]$, $[1, 10]$, $[2, 3]$, $[2, 6]$, $[2, 9]$, $[3, 6]$, $[4, 10]$, $[5, 7]$, $[5, 8]$, $[5, 10]$, $[7, 8]$. Indicați numărul minim de muchii care trebuie adăugate pentru ca graful obținut să aibă cel puțin un ciclu care să traverseze toate muchiile sale.
 - 1
 - 2
 - 3
 - 4

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este 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 ce se afișează dacă se citește valoarea 2186310. **(6p.)**
- b. Scrieți două numere din intervalul $[10^4, 10^5)$ care pot fi citite astfel încât, în urma executării algoritmului pentru fiecare dintre acestea să se afișeze două valori egale. **(6p.)**
- c. Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**
- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat structura `cât timp...execută` cu o structură repetitivă de alt tip. **(6p.)**
2. Variabila p memorează simultan gradul unui polinom (număr natural din intervalul $[0, 10^2)$) și toți coeficienții acestuia (numere reale). Știind că expresiile C/C++ de mai jos au ca valori gradul unui polinom, respectiv termenul liber al acestuia, scrieți definiția unei structuri cu eticheta `polinom`, care permite memorarea datelor despre un polinom, și declarați corespunzător variabila p . **(6p.)**
- `p.grad` `p.coeficient[0]`
3. Variabila s memorează un șir de cel mult 20 de caractere (litere mari și mici ale alfabetului englez). Declarați eventuale alte variabile necesare și scrieți o secvență de instrucțiuni în urma executării căreia se afișează pe ecran vocalele care NU apar în șirul menționat.
Se consideră vocale literele `a, e, i, o, u, A, E, I, O, U`.
Exemplu: dacă se citește textul `PREafrUMoASA` se afișează pe ecran, nu neapărat în această ordine, vocalele: `euOiI` **(6p.)**

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul `factori` are doi parametri, n și m , prin care primește câte un număr natural din intervalul $[1, 10^9]$. Subprogramul returnează numărul valorilor prime care apar la aceeași putere atât în descompunerea în factori primi a lui n , cât și în descompunerea în factori primi a lui m .
Scrieți definiția completă a subprogramului.
Exemplu: dacă $n=16500$ și $m=10780$, atunci subprogramul returnează 2 ($16500=2^2 \cdot 3 \cdot 5^3 \cdot 11$, $10780=2^2 \cdot 5 \cdot 7^2 \cdot 11$). **(10p.)**
2. Scrieți un program C/C++ care citește de la tastatură un număr natural, n ($n \in [2, 20]$), și construiește în memorie un tablou bidimensional cu n linii și n coloane, astfel încât fiecare linie a sa are proprietățile:
- elementul situat pe diagonala secundară are valoarea n ;
 - începând de la diagonala secundară, spre stânga, elementele formează un șir strict descrescător de numere consecutive, iar începând de la diagonala secundară, spre dreapta, elementele formează tot un șir strict descrescător de numere consecutive.
- Programul afișează pe ecran tabloul construit, fiecare linie a tabloului pe câte o linie a ecranului, cu elementele aflate pe aceeași linie separate prin câte un spațiu.
- Exemplu:** dacă $n=5$ se afișează pe ecran tabloul alăturat. **(10p.)**
- | | | | | |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| 2 | 3 | 4 | 5 | 4 |
| 3 | 4 | 5 | 4 | 3 |
| 4 | 5 | 4 | 3 | 2 |
| 5 | 4 | 3 | 2 | 1 |
3. Fișierul `bac.in` conține un șir de cel mult 10^6 numere întregi din intervalul $[-10^9, 10^9]$, separate prin câte un spațiu. Cel puțin un număr din șir este pozitiv.
Se cere să se afișeze pe ecran lungimea maximă a unei secvențe a șirului care fie începe, fie se încheie cu un număr pozitiv. O secvență este formată din termeni aflați pe poziții consecutive în șir, iar lungimea secvenței este egală cu numărul de termeni ai acesteia. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.
Exemplu: dacă fișierul conține numerele `-15 -7 4 -7 21 -5 -200 -26 52 -24 -7 -9 -20` pe ecran se afișează 11 (corespunzător secvenței `4 -7 21 -5 -200 -26 52 -24 -7 -9 -20`).
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
- b. Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**