

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

Varianta 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 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.

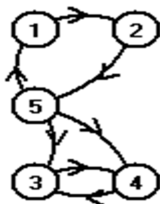
- Variabila x este de tip întreg și poate memora un număr din intervalul $[1000, 2000]$. Indicați valoarea minimă pe care o poate avea expresia C/C++ $(x-999)\%1000$ alăturată.

a. 0 b. 2 c. 999 d. 1000
- Subprogramul f este definit alăturat. Indicați ce se afișează în urma apelului de mai jos. $f(9)$;


```
void f(int x)
{
  if(x>=2)
  {
    if(x%2==0) cout<<0; | printf("0");
    f(x/2);
  }
  else cout<<7; | printf("7");
  cout<<x; | printf("%d",x);
}
```

a. 91 b. 942 c. 0071 d. 0071249
- Utilizând metoda backtracking, se generează toate meniurile care cuprind câte patru feluri distincte de mâncare din mulțimea {**aperitiv**, **desert**, **legume**, **paste**, **salată**, **supă**}, fiecare meniu respectând următoarele condiții:
 - dacă există aperitiv, este servit primul;
 - dacă există desert, este servit ultimul;
 - NU sunt servite legume și salată în același meniu;
 - dacă există paste și supă în același meniu, pastele NU sunt servite înainte de supă.
 Două meniuri sunt distincte dacă diferă prin cel puțin un fel de mâncare sau prin ordinea servirii acestora. Primele cinci meniuri generate sunt, în această ordine: (**aperitiv**, **legume**, **paste**, **desert**), (**aperitiv**, **legume**, **supă**, **desert**), (**aperitiv**, **legume**, **supă**, **paste**), (**aperitiv**, **paste**, **legume**, **desert**), (**aperitiv**, **paste**, **salată**, **desert**). Indicați al șaselea meniu generat.

a. (**aperitiv**, **salată**, **paste**, **desert**) b. (**aperitiv**, **salată**, **supă**, **desert**)

c. (**aperitiv**, **salată**, **supă**, **paste**) d. (**aperitiv**, **supă**, **legume**, **desert**)
- Un graf orientat cu 5 vârfuri, numerotate de la 1 la 5, este reprezentat alăturat. Indicați numărul de componente tare conexe ale grafului.
 

a. 1 b. 2 c. 3 d. 4
- Un arbore cu rădăcină are 20 de noduri, dintre care 10 noduri de tip "frunză". Indicați numărul maxim de noduri care au același "tată" în acest arbore.

a. 5 b. 7 c. 10 d. 15

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 .

- Scrieți ce se afișează dacă se citesc, în această ordine, numerele 5, 127, 2019, 1005, 7, 1900. **(6p.)**
- Dacă primul număr citit este 2, scrieți un set de numere distincte din intervalul $[10^3, 10^4]$ care pot fi citite în continuare astfel încât, în urma executării algoritmului, să se afișeze mesajul **niciunul**. **(6p.)**
- Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**

```

citește n (număr natural nenul)
m ← 0
pentru i ← 1, n execută
    citește x (număr natural)
    cât timp x%10 > [x/10]%10 execută
        x ← [x/10]
    m ← m+x
dacă m > 0 atunci scrie m
altfel scrie „niciunul”

```

- Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **pentru...execută** cu o structură repetitivă de alt tip. **(6p.)**

- Variabila c memorează date specifice unui cerc: coordonatele reale (abscisa și ordonata), în planul xOy , ale centrului cercului, precum și lungimea razei acestuia. Știind că expresiile C/C++ de mai jos au ca valori numere reale reprezentând datele specifice ale unui cerc, scrieți definiția unei structuri cu eticheta **figura**, care permite memorarea datelor precizate, și declarați corespunzător variabila c . **(6p.)**

$c.centru.x$ $c.centru.y$ $c.raza$

- Variabilele i și j sunt de tip întreg, iar variabila m memorează un tablou bidimensional cu 7 linii și 7 coloane, numerotate de la 0 la 6, cu elemente de tip **char**. Inițial, fiecare element memorează caracterul *****. Fără a utiliza alte variabile, scrieți secvența de instrucțiuni de mai jos, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila m să memoreze tabloul alăturat. **(6p.)**

```

for(i=0; i<7; i++)
    for(j=0; j<7; j++)
        .....

```

```

a b c d e f g
+ a b c d e f
+ + a b c d e
+ + + a b c d
+ + + + a b c
+ + + + + a b
+ + + + + + a

```

SUBIECTUL al III-lea

(30 de puncte)

- Subprogramul **multiplu** are un singur parametru, n , prin care primește un număr natural ($n \in [1, 10^4]$). Subprogramul returnează cel mai mic multiplu nenul al lui n cu proprietatea că este pătrat perfect. Scrieți definiția completă a subprogramului. **(10p.)**

Exemplu: dacă $n=72$ sau $n=144$, subprogramul returnează numărul 144 ($144=12^2$).

- Numim **cit** într-un text o secvență de caractere din acel text care începe cu un caracter $<$ și se termină cu un caracter $>$, celelalte caractere ale secvenței fiind diferite de $<$ și $>$. **(2p.)**

Un text de cel mult 100 de caractere (litere mici ale alfabetului englez, spații și caracterele $<$ și $>$) conține cel puțin un citat. Textul nu conține alte caractere $<$ și $>$ decât cele care mărginesc citatele, și oricare două citate nu au nici caractere $<$ și $>$ și nici alte caractere în comun.

Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și îl transformă în memorie prin înlocuirea tuturor literelor mici cuprinse în citate cu literele mari corespunzătoare, celelalte rămânând nemodificate, ca în exemplu. Programul afișează pe ecran textul obținut.

Exemplu: pentru textul **mai bine sa fii un <om de valoare> decat un <om de succes>** se afișează **mai bine sa fii un <OM DE VALOARE> decat un <OM DE SUCCES>** **(10p.)**

- Se numește **vârf** într-un șir de numere naturale un termen al șirului care este strict mai mare decât fiecare dintre cei doi termeni vecini cu el, aflați în șir pe poziția din stânga, respectiv din dreapta sa. **(2p.)**

Fișierul **bac.in** conține un șir de cel mult 10^6 numere naturale din intervalul $[0, 10^9]$, separate prin câte un spațiu. Se cere să se afișeze pe ecran vârful din șirul aflat în fișier pentru care valoarea absolută a diferenței dintre cei doi vecini ai săi este minimă. Dacă există mai multe astfel de numere, se afișează cel mai mare dintre ele, iar dacă nu există niciun vârf, se afișează pe ecran mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al spațiului de memorie utilizat.

Exemplu: dacă fișierul conține șirul 2 7 10 5 6 2 1 3 20 17 9 11 7 3 10 6 2 se afișează pe ecran 11

- Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
- Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**