

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

Testul 3

Filieră teoretică, profil real, specializare științe ale naturii

- 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.

**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.

- Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul memorat în variabila întregă  $x$  aparține reuniunii de intervale  $[-21, -2] \cup [2, 21]$ .
  - $x \geq -21 \ \&\& \ x \leq -2 \ \&\& \ x \geq 2 \ \&\& \ x \leq 21$
  - $!(x < -21 \ || \ x > -2) \ || \ !(x < 2 \ || \ x > 21)$
  - $x \geq -21 \ || \ x \leq -2 \ || \ x \geq 2 \ || \ x \leq 21$
  - $!(x < -21 \ \&\& \ x > 21 \ \&\& \ x > -2 \ || \ x < 2)$
- Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea  $x=48$ , se aplică metoda căutării binare, iar succesiunea de elemente a căror valoare se compară cu  $x$  pe parcursul aplicării metodei este 14, 24, 48. Elementele tabloului pot fi:
  - (2, 14, 7, 24, 12, 48, 50)
  - (4, 8, 9, 14, 16, 24, 48)
  - (14, 14, 24, 24, 48, 48)
  - (14, 24, 48, 14, 24, 48)
- Tabloul unidimensional  $A$  are elementele:  $A=(2, 20, 27, 36, 50)$ , iar în urma interclasării lui crescătoare cu tabloul unidimensional  $B$  se obține tabloul cu elementele (2, 3, 5, 8, 20, 27, 36, 45, 50, 63). Indicați elementele tabloului  $B$ , în ordinea apariției lor în acesta.
  - (1, -15, -19, 9, 13)
  - (2, 20, 27, 36, 50, 3, 5, 8, 45, 63)
  - (3, 27, 45, 63)
  - (63, 45, 8, 5, 3)
- Indicați o expresie C/C++ care are valoarea 1.
  - `floor(20.21)/20`
  - `floor(2020/21)`
  - `floor(1/2021)`
  - `floor(2020)/21`
- Indicați o secvență de instrucțiuni care realizează interschimbarea valorilor variabilelor întregi  $x$  și  $y$ .
  - `x=x-y; y=x-y; x=x-y;`
  - `x=x+y; y=x+y; x=x+y;`
  - `x=x+y; y=x-y; x=x-y;`
  - `x=x-y; y=x+y; x=x+y;`

**SUBIECTUL al II-lea** (40 de puncte)

- 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 întregă a numărului real  $c$ .
    - Scrieți ce se afișează dacă se citește numărul 250887. (6p.)
    - Scrieți două numere care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze valoarea citită. (6p.)
    - Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
    - Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat a doua structură repetitivă cu o structură repetitivă de alt tip. (6p.)
- ```
citește n
(număr natural nenul)
x ← 0; m ← 0; p ← 1
cât timp x < 10 execută
  cn ← n
  cât timp cn ≠ 0 execută
    c ← cn % 10; cn ← [cn / 10]
    dacă c = x atunci
      m ← c * p + m; p ← p * 10
  x ← x + 2
scrie m
```

2. Pentru un exponat dintr-un muzeu se memorează un cod (număr natural cu cel mult cinci cifre) și valoarea evaluată a acestuia (număr real). Variabilele `cod1` și `valoare1` memorează codul, respectiv valoarea evaluată a unui exponat, iar variabilele `cod2` și `valoare2` memorează codul, respectiv valoarea evaluată a unui alt exponat. Știind că cele două exponate au valori diferite, declarați variabilele `valoare1` și `valoare2` și scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran, separate printr-un spațiu, codul exponatului cu valoarea cea mai mare și valoarea medie evaluată a celor două exponate. **(6p.)**
3. Toate variabilele din secvența de mai jos sunt de tip `char`.

Scrieți ce se afișează în urma executării secvenței.

**(6p.)**

```
c1='m';c2='u';c3='z';c4='e';  
c1=c1+1;  
cout<<c2<<c1<<endl; | printf("%c%c\n",c2,c1);  
cout<<c3<<c4<<c2; | printf("%c%c%c",c3,c4,c2);
```

### **SUBIECTUL al III-lea**

**(30 de puncte)**

1. Se citește un număr natural nenul,  $n$  și se cere să se scrie suma divizorilor pozitivi ai lui  $n$  care nu sunt primi. Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate. **(10p.)**  
**Exemplu:** pentru  $n=12$  se scrie 23 ( $23=1+4+6+12$ ).
2. O persoană poate rezerva mai multe locuri într-o sală de spectacole, dar cel mult câte unul în aceeași jumătate a unui rând. Configurația primului rând al sălii este memorată ca un tablou unidimensional, în care fiecare element corespunde câte unui loc, în ordine de la stânga la dreapta. Locurile libere sunt notate cu 0, iar cele rezervate sunt notate cu numere naturale nenule, astfel încât toate locurile rezervate de o persoană sunt notate cu același număr, iar cele rezervate de persoane distincte sunt notate cu numere distincte.  
Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $n \in [2, 20]$ ), apoi  $2 \cdot n$  numere naturale din intervalul  $[0, 400]$ , elemente ale tabloului unidimensional reprezentând configurația primului rând al sălii. Programul verifică dacă există persoane care au obținut locuri atât în jumătatea din stânga, cât și în jumătatea din dreapta a acestui rând, și afișează pe ecran numerele cu care sunt notate aceste locuri, ca în exemplu. Numerele sunt afișate într-o ordine oarecare, separate prin câte un spațiu, iar dacă nu există astfel de locuri se afișează pe ecran mesajul **nu exista**.  
**Exemplu:** pentru  $n=5$  și tabloul  $(\underline{7}, \underline{0}, \underline{5}, \underline{0}, \underline{2}), (\underline{0}, \underline{2}, \underline{1}, \underline{5}, \underline{3})$  se afișează pe ecran, nu neapărat în această ordine, numerele 2 5 **(10p.)**
3. Fișierul `bac.in` conține un șir de cel mult  $10^6$  numere naturale din intervalul  $[10, 10^9]$ , separate prin câte un spațiu. Se cere să se afișeze pe ecran toate valorile lui  $z$  cu proprietatea că în șir se află un număr maxim de termeni cu cifra zecilor egală cu  $z$ . Cifrele afișate sunt separate prin câte un spațiu. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.  
**Exemplu:** dacă fișierul conține numerele 726 378 98 177 20 49 128 859 679 251 pe ecran se afișează cifrele de mai jos (sunt câte 3 termeni care au cifra zecilor 2, respectiv 7):  
2 7  
a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**  
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**