

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

Testul 7

Filieră teoretică, profil real, specializare științe ale naturii

- 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.

I. TÊTEL

(20 pont)

Minden item esetében 1-től 5-ig írja a vizsgalapra a helyes válasznak megfelelő betűt. Minden helyes válasz 4 pontot ér.

1. A
 $!(x < 2001) \ \&\& \ (x \leq 2002) \ || \ (x \geq 2019) \ \&\& \ !(x > 2020)$
C/C++ kifejezés értéke akkor és csakis akkor 1, ha az x valós változó értéke eleme ennek a halmaznak:
a. $\{2001\} \cup [2002, 2019] \cup \{2020\}$ b. $[2001, 2002] \cup \{2019\} \cup \{2020\}$
c. $[2001, 2002] \cup \{2019, 2020\}$ d. $[2001, 2002] \cup [2019, 2020]$
2. Annak leellenőrzésére, hogy a $(2, 4, 7, 8, 14, 17, 30)$ egydimenziós tömbben megtalálható-e az x érték, a bináris keresés módszerét használjuk. Tudva, hogy a módszert alkalmazva, az x értéket a tömb két elemével hasonlítottuk össze, az x két lehetséges értéke:
a. 4, 17 b. 7, 14 c. 2, 30 d. 2, 4
3. Az $A = (2, 6, 7, 10, 16)$ egydimenziós tömb, amelyre a B tömbbel növekvő sorrendben való összefésülést alkalmazva, eredményként a $(2, 5, 6, 7, 9, 10, 16, 16, 27, 49)$ tömböt kapjuk. Add meg a B tömb elemeit, előfordulási sorrendjüket figyelembe véve.
a. $(2, 6, 7, 10, 16, 5, 9, 16, 27, 49)$ b. $(2, 3, 6, 3, 7, 9, 10, 17, 16, 33)$
c. $(49, 27, 16, 9, 5)$ d. $(33, 17, 9, 3, 3)$
4. Az x , y és i változók egész típusúak és értékeik nullától különbözőek. Add meg az utasítást, amely a mellékelttel egyenértékű átalakítást eredményez.
a. $x = 2020 / y;$ b. $x = y / 2020;$ c. $x = y;$
for ($i = 2; i \leq 2020; i++$)
 $x = x * y;$ d. $x = 2020;$
for ($i = 2; i \leq y; i++$)
 $x = x * 2020;$
5. Jelöljete meg egy utasítást, amely az x és y egész változók értékeinek felcserélését valósítja meg.
a. $x = x - y; \ y = x - y; \ x = x - y;$ b. $x = x + y; \ y = x + y; \ x = x + y;$
c. $x = x + y; \ y = x - y; \ x = x - y;$ d. $x = x - y; \ y = x + y; \ x = x + y;$

II. TÊTEL

(40 pont)

1. Adott a mellékelt algoritmus pszeudokódban.
Az $a \% b$ jelöli az a természetes szám b nem nulla természetes számmal való osztási maradékát, és $[c]$ a c valós szám egész részét
a. Írjátok le mit ír ki, ha a beolvasott szám 253387. (6p.)
b. Írjátok le a $[10, 10^2)$ intervallumból, a legkisebb és legnagyobb számot, amelyeket beolvasva, a kiírt eredmény 0 lesz. (6p.)
c. Írjátok az adott algoritmusnak megfelelő C/C++ programot. (10p.)
d. Írjátok pszeudokódban az eredetivel egyenértékű algoritmust, az első ismétlődő struktúrát megfelelőképpen lecserélve egy más típusú ismétlődő struktúrára. (6p.)
- ```
beolvas n
(nem nulla természetes szám)
x ← 1; m ← 0; p ← 1
amíg x < 10 végezd el
 cn ← n
 amíg cn ≠ 0 végezd el
 c ← cn % 10; cn ← [cn / 10]
 ha c = x akkor
 m ← c * p + m; p ← p * 10
 x ← x + 2
kiír m
```

2. Egy könyvtári könyvvel kapcsolatban tároljuk a leltári számot (természetes szám, legtöbb öt számjegyű) és a könyv árát (valós szám). Az **inv1** és **pret1** változók egy könyv leltári számát és árát tárolják, míg az **inv2** és **pret2** egy másik könyv leltári számát és árát tárolják.

Értelmezték a **pret1** és **pret2** változókat és írjatok egy utasítássort, amelyet végrehajtva, megjelenik a képernyőre kiírva, egy szóközzel elválasztva, az olcsóbb könyv leltári száma, majd a két könyv árainak átlaga. (6p.)

3. A mellékelt kódban minden változó típusa **char**. Adjátok meg mit ír ki a kódrészlet. (6p.)
- ```
c1='s';c5='o';c6='u';
cout<<c5<<c6<<endl; | printf("%c%c\n",c5,c6);
c1=c1-1; c2=c1-3;
c3=c1+1; c4=c1+3;
cout<<c1<<c2<<c3<<c4; | printf("%c%c%c%c",c1,c2,c3,c4);
```

III. TÉTEL

(30 pont)

1. Két természetes számot olvassunk be, **n** ($n \geq 2$) és **p** és írassuk ki azt a hatványkitevőt, amelyen a **p** szám szerepel az **n** szám prímtényezőre bontásában, ha a **p** prímszám, vagy a -1 értéket, ellenkező esetben.

Írjátok meg pszeudokódban, a megfogalmazott feladat megoldására szolgáló algoritmust.

Példa: ha **n=80** és **p=2**, a kiírt érték **4** ($80=2^4 \cdot 5$).

(10p.)

2. Egy **k** érték polarizál egy számsort, ha létezik két tag a sorból, amelyeknek az értéke azonos, az első a sor első feléből, a másik a sor második feléből származzon.

Írjatok egy C/C++ programot, amely a billentyűzetről természetes számokat olvas be az **[1,20]** intervallumból: **n** és egy $2 \cdot n$ darab elemmel rendelkező számsor, amelyre jellemző, hogy az első **n** darab szám is és az utolsó **n** szám is páronként különbözőek. A program írja ki az értékeket, amelyek polarizálhatják a beolvasott sort.

A kiírt értékek sorrendje tetszőleges, egy-egy szóközzel elválasztva, ha nem létezik ilyen érték, a képernyőre a **nepolarizat** szöveg íródjon ki.

Példa: ha **n=4** és a tömb (3,2,6,5,5,1,6,8) a képernyőre kiírt értékek, nem feltétlenül ebben a sorrendben, az 5 6 számok. (10p.)

3. A **bac.txt** állomány tartalmaz egy számsort, amelynek legföljebb 10^6 darab, egész szám eleme van a $[-10^3, 10^3]$ intervallumból, egy-egy szóközzel elválasztva. A képernyőre írja ki a program azt a legkisebb összeget, amit az állományban található sorban, egymásután elhelyezkedő, negatív elemek összeadásából megkaphat. Tervezzetek a felhasznált memória és a végrehajtási idő szempontjából hatékony algoritmust.

Példa: ha a **bac.txt** állomány tartalma -4 6 -7 -2 -1 -4 10 3 -9 -2 2

a képernyőre kiírt érték a -14

a. Írjátok a tervezett algoritmusnak megfelelő C/C++ programot. (8p.)

b. Írjátok le saját szavaitokkal a használt algoritmust, és indokoljátok annak hatékonyságát. (2p.)