

Examenul național de bacalaureat 2023
Proba E. d)
Informatică
Limbajul C/C++

Varianta 7

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 arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

I TÉTEL (20 pont)

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

1. Adja meg a számokat, melyeket **x** és **y** egész típusú változóknak tárolhat, úgy, hogy a mellékelt C/C++ kifejezés értéke 1 legyen. $y \% x - (x / y) * 3 != 0$
- a. **x=25** és **y=75** b. **x=25** és **y=15** c. **x=15** és **y=0** d. **x=10** és **y=30**
2. A **f** alprogram mellékelve van megadva. Adja meg az **f(3,2)** értékét.
- ```
int f(int x, int y)
{
 int z;
 if (y==0) return 1;
 z=f(x,y/2);
 if (y%2!=0) return z*z*x;
 return z*z;
}
```
- a. 1      b. 2      c. 9      d. 18
3. Az **s** változóban legtöbb 20 karakter érhető el. Adja meg az **s**-ben elérhető sort a mellékelt művelet sor végrehajtása után. `strcpy(s, "2019");`  
`strcpy(s+3, "23");`
- a. 20123      b. 201923      c. 202223      d. 2023
4. Használva a backtracking algoritmus módszerét, generálva lesz az összes parfüm, melyet 3 különböző esszencia keverékeként kapunk a {**bergamotă**, **cireș**, **iris**, **lămâie**, **salcâm**} halmazból. Két parfüm különböző, ha legalább egy esszenciában különböznek. Az első négy kapott megoldás, ebben a sorrendben a következők: (**bergamotă**, **cireș**, **iris**), (**bergamotă**, **cireș**, **lămâie**), (**bergamotă**, **cireș**, **salcâm**) és (**bergamotă**, **iris**, **lămâie**). Adja meg az utolsó előtti megoldást.
- a. (**cireș**, **iris**, **salcâm**)      b. (**cireș**, **lămâie**, **salcâm**)  
c. (**lămâie**, **iris**, **salcâm**)      d. (**iris**, **lămâie**, **salcâm**)
5. Egy irányítatlan gráfnak 5 csomója van, sorszámozva 1-től 5-ig, a mellékelt táblában lévő fokszámokkal. Adjon meg egy lehetséges értékpárost **x** és **y**-nak.
- |         |   |   |   |   |   |
|---------|---|---|---|---|---|
| Csomó   | 1 | 2 | 3 | 4 | 5 |
| Fokszám | 2 | x | 3 | 3 | y |
- a. 0 és 4      b. 1 és 5      c. 2 és 3      d. 3 és 3

## II TÉTEL

(40 punct)

### 1. A mellékelt algoritmus pszeudokódban van megadva.

Jelölve van  $[c]$  -vel a  $c$  valós szám egész része.

- Írja le a megjelenített értéket, ha sorban a 3, 746, 82, 3067, 67, 78, 178 számok lesznek beolvasva. (6p.)
- Ha az  $n$ -nek a 2 számot olvassuk, írjon egy természetes számsort a  $[0, 9]$  intervallumból, mely folyamatosan olvasható abban a sorrendben, úgy, hogy az algoritmus végrehajtásakor a 0 érték legyen megjelenítve. (6p.)
- Írjon az adott algoritmusnak megfelelő C/C++ programot. (10p.)
- Írjon pszeudokódban egy algoritmust, mely egyenértékű a megadottal, helyettesítve az első **amíg...végezd el** struktúrát egy **minden...végezd el** ismétlődő struktúrával. (6p.)

```

olvas n (nemnulla természetes szám)
x ← 0
amíg n > 0 végezd el
 olvas a, b (természetes számok)
 c ← a; p ← 1
 amíg a > 9 végezd el
 a ← [a/10]; p ← p * 10
 a ← a * p + b
 ha a ≠ c akkor
 x ← x + 1
 n ← n - 1
kiír x

```

- Egy gyökeres fában egy csomó az  $x$  szinten található, ha az az elemi lánc, melynek egyik vége az adott csomóban van és a másik vége a gyökérben,  $x$  hosszúságú. A 0 -ik szinten egy csomó található (a gyökér).  
Egy fa 6 csomóval, 1 -től 6 -ig sorszámozva,  $[1, 2]$ ,  $[2, 3]$ ,  $[2, 6]$ ,  $[3, 4]$ ,  $[3, 5]$  élekkel adott. Írja le azokat a csomókat, melyek gyökek lehetnének, úgy, hogy a kapott fák minimális szintűek. (6p.)
- A  $p$  változó, mely mellékelten van deklarálva, egy termék jellemzőit tárolja: denumire (megnevezés) és pret (ár). Írja le azt a C/C++ műveletsort, mellyel az  $a$  változó az adott termék megnevezésének első kezdőbetűjét tárolja, ha ennek az ára szigorúan kisebb, mint 100, vagy az  $*$  karaktert, ellenkezőleg. (6p.)

```

struct produs
{
 char denumire[20];
 int pret;
};
p;
char a;

```

## III TÉTEL

(30 punct)

- A **DNPI** alprogramnak egy paramétere van, az  $n$ , melyen keresztül egy természetes számot kap ( $n \in [1, 10^9]$ ), és kiírja a képernyőre, egy-egy szóközzel elválasztva, az  $n$  összes pozitív páratlan osztóját, melyek **NEM** prímek. Írja le a teljes alprogramot.

**Például:** ha  $n=90$ , a képernyőre kiírt értékek, nem feltétlenül ebben a sorrendben, a

1 9 15 45

(10p.)

- Egy játék téglalap alakú táblát használ, melyen egyforma cellákat jelöltek meg,  $m$  sorra és  $n$  oszlopra bontva. Minden cellába egy természetes szám íródott.  
Négy cellát, melyek két egymást követő soron, két egymást követő oszlopon helyeztek el, azzal a tulajdonsággal, hogy a cellába írt értékek összege  $p$ ,  $p$  értékű **négyzetnek** nevezünk. A játék célja egy maximális értékű négyzet meghatározása.

Írjon C/C++ programot, amely beolvas a billentyűzetről két természetes számot,  $m$  és  $n$ , a  $[2, 20]$  intervallumból, aztán egy  $m$  soros és  $n$  oszlopos kétdimenziós tábla elemeit, természetes számokat a  $[0, 10^4]$  intervallumból, a játéktáblára írt számokat, a cellák elhelyezésének megfelelő sorrendjében. A program meghatározza a tábla egy maximális értékű négyzetét és kiírja a képernyőre ezt az értéket.

**Például:** ha  $m=5$ ,  $n=4$  és a mellékelt tábla, a képernyőre kiírt érték 36, az ábrán kiemelt négyzetnek megfelelően. (10p.)

|   |    |    |    |
|---|----|----|----|
| 1 | 1  | 2  | 30 |
| 3 | 10 | 1  | 2  |
| 1 | 13 | 12 | 1  |
| 1 | 2  | 3  | 1  |
| 7 | 1  | 20 | 1  |

- A **date.in** első során két természetes szám található az  $[1, 10^6]$  intervallumból,  $m$  és  $n$ , valamint a következő két soron természetes számok a  $[0, 10^2]$  intervallumból: a második soron egy **A** sor,  $m$  számból, valamint a harmadik soron egy **B** sor,  $n$  számból. Az azonos soron található számokat egy-egy szóköz választja el egymástól. Írja ki a képernyőre a  $(pa, pb)$  ( $pa \in [1, m]$ ,  $pb \in [1, n]$ ) formájú párok maximális számát, azzal a tulajdonsággal, hogy az **A** sor  $pa$  pozícióján található eleme azonos értékű a **B** sor  $pb$  pozícióján található elemével, és hogy minden A, valamint B sornak megfelelő pozíció, legtöbb egy párban jelenik meg, ahogy a példa mutatja. Tervezzon futási idő szempontjából hatékony algoritmust.

**Például:** ha az állomány a mellékelt számokat tartalmazza, a képernyőre kiírt érték 6

(például, az  $(1, 1)$ ,  $(2, 9)$ ,  $(4, 2)$ ,  $(5, 5)$ ,  $(6, 6)$ ,  $(7, 7)$  párok vagy az  $(1, 2)$ ,  $(2, 9)$ ,  $(4, 1)$ ,  $(5, 7)$ ,  $(6, 8)$ ,  $(8, 5)$  párok).

a. Írja le természetes nyelven a tervezett algoritmust, igazolva annak hatékonyságát. (2p.)

b. Írja meg a tervezett algoritmusnak megfelelő C/C++ programot. (8p.)

|   |   |
|---|---|
| 8 | 9 |
| 1 | 0 |
| 4 | 1 |
| 5 | 3 |
| 5 | 5 |
| 1 | 1 |
| 1 | 7 |
| 5 | 3 |
| 5 | 3 |
| 0 |   |