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

Testul 6

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)**

Írja a vizsgalapra az 1 és 5 közötti feladatokra adott helyes válaszoknak megfelelő betűt. Mindegyik helyes válasz 4 pontot ér.

1. A következő C/C++
`(x>=18) && !(x<19 || x>20) && (x<=21)`
kifejezés értéke akkor és csakis akkor 1 ha az x egész változó a következő intervallumhoz tartozik:

a. `[18,20]` **b.** `[18,21]` **c.** `[19,20]` **d.** `[19,21]`

2. A backtracking módszert használva legeneráljuk öt cserje összes ültetési módozatát a következő halmazból {**caprifoi**, **iasomie**, **liliac**, **tamarix**, **scumpie**}. Két megoldás különbözik, ha a cserjék sorrendje különbözik. Az így kapott első négy megoldás: (**caprifoi**, **iasomie**, **liliac**, **tamarix**, **scumpie**), (**caprifoi**, **iasomie**, **liliac**, **scumpie**, **tamarix**), (**caprifoi**, **iasomie**, **tamarix**, **liliac**, **scumpie**), (**caprifoi**, **iasomie**, **tamarix**, **scumpie**, **liliac**). Jelölje be az utolsó előtti megoldást.

a. **(scumpie**, **tamarix**, **caprifoi**, **iasomie**, **liliac)**

b. **(scumpie**, **tamarix**, **caprifoi**, **liliac**, **iasomie)**

c. **(scumpie**, **tamarix**, **liliac**, **caprifoi**, **iasomie)**

d. **(scumpie**, **tamarix**, **liliac**, **iasomie**, **caprifoi)**

3. Az **A** és **B** változók egy-egy pont koordinátáit tárolják (**x** abszcissza, **y** ordináta) **xOy** koordinátarendszerben. Jelölje meg azt a C/C++ kifejezést amelyik értéke 1, akkor és csakis akkor ha az **A** és **B** végpontok által meghatározott szakasz metszi az **Oy** tengelyt a koordinátarendszerben.

```
struct punct
{ int x,y;
} A,B;
```

a. `A.x*B.x<=0` **b.** `A(x)*B(x)<=0` **c.** `x.A*x.B<=0` **d.** `punct(A,B).y==0`

4. Egy gyökeres fában egy csúcspont az **x** szinten helyezkedik el, ha az illető csúcspont és a gyökér közötti elemi lánc hossza **x**. A 0-ik szinten egy csúcspont (gyökér) van.
Az adott gyökeres fának 8 csúcspontja van, a csúcspontok 1-től 8-ig vannak sorszámozva, a fa élei a következők: **[1,3]**, **[1,7]**, **[1,8]**, **[2,4]**, **[3,5]**, **[3,6]**, **[4,5]**. Tudva azt, hogy a fa gyökere a 5-ös sorszámú csúcspont, jelölje be a 2-ik szinten lévő csúcspontokat az adott fában.

a. `7,8` **b.** `7,4` **c.** `1,4,6` **d.** `1,2,6`

5. Egy 5 csúcspontot tartalmazó, 1-től 5-ig sorszámozott irányított gráfban az élek a következők **(2,5)**, **(3,1)**, **(5,3)**, **(5,4)**. Jelölje be, hogy minimálisan hány élt kell a gráfhoz hozzáadni, hogy a gráf erősen összefüggő legyen.

a. `1` **b.** `2` **c.** `3` **d.** `4`

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

1. **Adott a mellékelt algoritmus pszeudokódban.**

a. Mit fog kiírni az algoritmus végrehajtás után, ha az **n**-be beolvasott érték 7. **(6p.)**

b. Adjon meg két olyan értéket a **[10,10²)** intervallumból, amelyek beolvasása és az algoritmus végrehajtása után 10-et fog kiírni az algoritmus. **(6p.)**

c. Írja meg az adott algoritmusnak megfelelő C/C++ programot. **(10p.)**

d. Írja meg az adott algoritmussal egyenértékű algoritmust pszeudokódban úgy, hogy az **amíg...végezd el** ciklust helyettesítsük egy **minden...végezd el** ciklussal. **(6p.)**

```
beolvas n (nem nulla természetes szám)
  nr←0; i←1
┌amíg i≤n végezd el
│ x←0; y←1; j←1
│ ┌amíg j<i végezd el
│ │ r←2*x-y; x←y; y←r
│ │ j←j+1
│ └■
│ i←i+1
│ ┌ha y>0 akkor
│ │ nr←nr+1
│ └■
└■
kiír nr
```

2. Adott az f alprogram. Adjon meg két természetes számot az **[1,10]** intervallumból, amelyeket, ha az x1 és x2 egész változók értékül kapják, akkor az **f(10,x1)** visszatérési értéke **5**, és az **f(x2,10)** visszatérési értéke **1** lesz. **(6p.)**

```
int f(int a, int b)
{ if(a>b) return a/b+f(a-b,b);
  if(a<b) return b/a+f(a,b-a);
  return 1;
}
```

3. Az i és j változók egész típusúak, az **a** változó egy kétdimenziós tömb 4 sorral és 5 oszloppal. A tömb sorai és oszlopai 0-tól vannak sorszámozva, minden eleme egész és kezdeti értékük 0. Anélkül, hogy az adott változókon kívül más változókat is használnánk, írjon egy utasítássorozatot, amelynek a végrehajtása után az **a** kétdimenziós tömb a mellékelt táblázatot tartalmazza. **(6p.)**

```
16 17 18 19 20
11 12 13 14 15
 6  7  8  9 10
 1  2  3  4  5
```

SUBIECTUL al III-lea **(30 de puncte)**

1. A **prodprim** alprogramnak két paramétere van:

- **n**, természetes szám (**n∈[2,10⁹]**);
- **p**, amelyen keresztül visszaadja az n szám prím osztóinak a szorzatát.

Írja meg a teljes definícióját az alprogramnak.
Példa: ha **n=2000**, akkor a hívás után a **p=10**, mivel **2000=2⁴·5³**. **(10p.)**

2. Adott egy maximum 100 karakterből álló szöveg, amelyben a szavak az angol abc kisbetűiből állnak és egy-egy szóközzel vannak elválasztva. Írjon egy **C/C++** programot, amely a fent említett szöveget beolvassa a billentyűzetről és kiírja a képernyőre egy-egy sorba azokat a szavakat amelyekben a magánhangzók száma szigorúan kisebb mint a mássalhangzók száma. Ha nincs ilyen szó, akkor írja ki, hogy **nu exista**. Magánhangzónak tekintjük a következő betűket **a**, **e**, **i**, **o**, **u**.

Példa: a következő szövegre **ei au plantat tamarix ea a adus iasomie**
Kiírja a képernyőre nem feltétlenül ebben a sorrendbe a mellékelt szavakat. **(10p.)**

```
plantat
tamarix
```

3. Beolvassunk a billentyűzetről két természetes számot **p1-t** és **p2-t** az **[1,81]** intervallumból, írja ki a **bac.out** fájlba az összes olyan számot, amely pontosan 7 számjegyű és az első két számjegy szorzata p1, a következő 3 számjegy egyenlő egymással és az utolsó két számjegy szorzata p2. A fájlban a számok csökkenő sorrendbe kell szerepeljenek, minden sorba egy szám kell kerüljön. Tervezzen meg egy hatékony algoritmust a memória felhasználás és a futási idő szempontjából.
Példa: ha **p1=12** és **p2=8**, akkor **26333<u>24</u>** és **34000<u>18</u>** két szám a **160** számból amelyek a kért tulajdonsággal rendelkeznek. (**2·6=3·4=12** și **2·4=1·8=8**).

a. Írjon egy C/C++ programot a megtervezett algoritmusnak megfelelően. **(8p.)**

b. Írja le magyarul a megtervezett algoritmust és indokolja meg ennek a hatékonyságát. **(2p.)**