



Felvételi, 2017. július

– Alapképzés, informatika vizsga –

Hivatalból 1 pont jár. Munkaidő 3 óra.

Megjegyzések:

- A pszeudokód algoritmusban
 - az „=” értékadást jelent
 - Az $x[i][j]$ hivatkozás az x tömb i -edik sorának j -edik elemét jelenti.
 - A „minden $i = 1, n, 1$ végezd” jelentése: minden $i = 1, 2, 3, \dots, n$ értékre végezd el...
 - A „minden $i = n, 1, -1$ végezd” jelentése: minden $i = n, n-1, n-2, \dots, 1$ értékre végezd el...
- Minden egész érték tárolható integer/int típusú változóknban.
- A bemeneti adatok helyesnek tekinthetők.
- A programokat megjegyzésekkel kell ellátni!
- Törekedj hatékony megoldásokra!

1. (1 pont) Mit ír ki az alábbi pszeudokód algoritmus, ha az n, i, j változók értékei rendre 8, 2, 7?
Adjunk példát olyan n, i, j bemenetre, amelyre az algoritmus 8-at ír ki!

```
p = 0
ha i-2 < 1 VAGY j-1 < 1 akkor
    p = p + 1
ha i-2 < 1 VAGY j+1 > n akkor
    p = p + 1
ha i+2 > n VAGY j-1 < 1 akkor
    p = p + 1
ha i+2 > n VAGY j+1 > n akkor
    p = p + 1
ha i-1 < 1 VAGY j-2 < 1 akkor
    p = p + 1
ha i-1 < 1 VAGY j+2 > n akkor
    p = p + 1
ha i+1 > n VAGY j-2 < 1 akkor
    p = p + 1
ha i+1 > n VAGY j+2 > n akkor
    p = p + 1
Kiír 8 - p
```

2. (1 pont) Mi lesz az $a[1..3][1..3]$ kétdimenziós tömb tartalma az alábbi pszeudokód algoritmusrészlet nyomán, ha előzőleg minden eleme 0-t tárolt?

```

k = 1
minden j = 1, 2, 1 végezd
|   a[1][j] = k
|   k = k + 1
■
minden i = 1, 2, 1 végezd
|   a[i][3] = k
|   k = k + 1
■
minden j = 3, 2, -1 végezd
|   a[3][j] = k
|   k = k + 1
■
minden i = 3, 2, -1 végezd
|   a[i][1] = k
|   k = k + 1
■

```

3. (1 pont) Mi lesz az $a[1..5]$ és $b[1..5]$ öt-elemű tömbök tartalma az alábbi pszeudokód algoritmusrészlet nyomán, ha előzőleg az a tömb a 3, 2, 1, 5, 4, a b pedig a 0, 0, 0, 0, 0 értékeket tárolta?

```

minden i = 1, 5, 1 végezd
|   b[a[i]] = b[a[i]] + 1
■

```

4. (1 pont) Írjunk Pascal vagy C/C++ programot, amely kiírja, egymás alá a képernyőre, az egymást követő évszámokat, kezdve 2000-rel, és befejezve 2017-tel.

5. (1 pont) Írjunk Pascal vagy C/C++ programot, amely beolvas a billentyűzetről egy n természetes számot ($n \neq 2017$), majd kiírja a képernyőre a természetes számokat n -től 2017-ig. Ha n kisebb, mint 2017, akkor egy növekvő, ha nagyobb, akkor egy csökkenő számsorozatot várunk.

6. (1 pont) Egy egész számokat tartalmazó tömb egy halmaz (nincsenek ismétlődő elemek; az elemek sorrendje nem fontos) elemeit tárolja. Írjunk Pascal vagy C/C++ függvényt, amely a halmazból töröl egy adott számot, amennyiben az eleme a halmaznak. A függvény paraméterei legyenek a halmazt tároló tömb, a halmaz mérete, illetve a törlendő szám, és térítse vissza a frissített halmaz méretét.

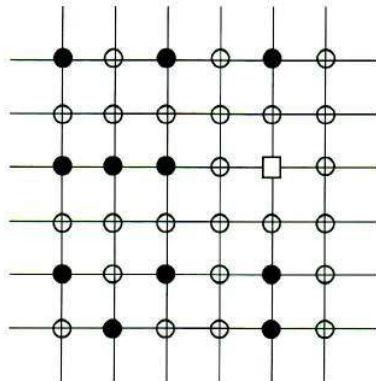
7. (1 pont) Írjunk Pascal vagy C/C++ függvényt, amely a paraméterként kapott n elemű tömből eldönti, hogy az $\{1, 2, 3, \dots, n\}$ halmaz egy permutációját tartalmazza-e.

8. (1 pont) Egy bementi állományban adott az n ($2 \leq n \leq 10$) érték és egy $n \times n$ méretű természetes számokat tartalmazó mátrix. Írjunk Pascal vagy C/C++ programot, amely beolvassa a mátrixot, majd cirkulárisan permutálja a tömb sorait, az i -edik sort i -szer ($i=1, 2, \dots, n$) (cirkulárisan permutálni *egyszer* azt jelenti, hogy az illető sor minden eleme eggyel jobbra, az utolsó elem pedig az első elem helyére lép). Az eredmény mátrixot a képernyőre írássuk ki.

9. (1 pont) Az ábrán látható $n \times n$ ($1 \leq n \leq 999$) méretű rács rácspontjaiban elhanyagolható vastagságú fák találhatók. Az (i, j) pozícióban található fát lecseréljük egy elhanyagolható vastagságú erdőszre. Írjunk Pascal vagy C/C++ függvényt, amely paraméterként megkapja az n , i , j értékeket és visszatéríti, hogy hány fát lát az erdősz.

Példa: az $(n = 6, i = 3, j = 5)$ bemenetre a helyes válasz 24.

(a rácsvonalak 6x6 metszéspontjában fák vannak, kivéve a $(3, 5)$ pozíciót, ahol az erdősz tartózkodik; az erdősz négyzet, a látott fákat átlátszó körök, a takartakat pedig satírozott körök jelölik)



Informatika tétel, 2017 július – Javítókulcs

Hivatalból – 1 pont

1. Feladat (1 pont)

- Mit ír ki? – **4** – 0,5 pont
- Adjunk példát ... – *például* **8, 4, 4** – 0,5 pont

2. Feladat (1 pont)

1	2	3
8	0	4
7	6	5

3. Feladat (1 pont)

a

3	2	1	5	4
---	---	---	---	---

 – 0,2 pont

b

1	1	1	1	1
---	---	---	---	---

 – 0,8 pont

4. Feladat (1 pont)

- Helyes ciklus – 1 pont

5. Feladat (1 pont)

- Helyes elágazás/feltétel – 0,2 pont
- Növekvő sorozat – 0,4 pont
- Csökkenő sorozat – 0,4 pont

6. Feladat (1 pont)

- Elem keresése – 0,5 pont
 - Ha nem optimális (találat után a keresés folytatása) – 0,4 pont
- Törlés – 0,4 pont
 - Ha nem optimális (egynél több elem felülírása) – 0,2 pont
- Halmaz méretének csökkentése – 0,1 pont

7. Feladat (1 pont)

- Optimális algoritmus (lineáris, lásd 3. feladat) – 1 pont
- Nem optimális algoritmus (négyzetes) – 0,8 pont

8. feladat (1 pont)

- Beolvasás – 0,1 pont
- Permutálások – 0,8 pont
 - Ciklusszervezés – 0,1 pont
 - Egy sor permutálása – 0,7 pont

Informatika tétel, 2017 július – Javítókulcs

- Optimális algoritmus (lineáris) – 0,7 pont
- Nem optimális algoritmus (négyzetes) – 0,5 pont
- Kiírás – 0,1 pont

9. feladat (1 pont)

- $n = 1$ helyes kezelése – 0,1 pont
- erdész sorának és oszlopának helyes kezelése – 0,3 pont
- általános eset – 0,6 pont
 - erdész pozíció: $((i_{erdész}, j_{erdész})$; fa pozíció: (i_{fa}, j_{fa})
 - $|i_{erdész} - i_{fa}|$ és $|j_{erdész} - j_{fa}|$ relatív prímek