

Testare finală - Grupa de excelență CNILC2 – semigrupa 1 An școlar 2015-2016

Partea I (20p)

Se acorda **10 p** din oficiu

1. Fie șirul de numere 1, 12, 123, 1234, ...12345678910,.... Considerându-se primele N numere ale acestui șir afișați numărul de numere care sunt divizibile cu 3, dacă $N=3564$ **(10p)**

2.

- 1 a) Ce valori vor fi afișate pentru $n=7$ și valorile 3, 5, 6, 4, 8,
2 9, 2? Dar pentru $n=7$ și valorile 3, 5, 6, 7, 8, 9, 23? **(2p)**
3
4
5 b) Dați exemplu de valori pentru care se va afișa șirul de
6 valori 8 5 3 12. **(2p)**
7
8
9 c) Ce semnificație dați valorii variabilei nr ? **(3p)**
10
11 d) Considerând o valoare oarecare pentru n , cum trebuie să
12 fie șirul valorilor introduse pentru ca la final să se afișeze tot
13 cele n valori? **(3p)**
14
15
16
17
18
19
20

```
intreg a[100], max, i, n;
citeste n, a[1];
nr ← 1;

pentru i ← 2, n executa
    citeste x;
    k ← 1;
    cat timp (x ≤ a[k] and (k ≤ nr)
        executa
            k ← k + 1;
    daca k = nr + 1 atunci
        nr ← nr + 1;
    a[k] ← x;

pentru i ← 1, nr executa
    scrie a[i];
```

Partea II (70p)

- Să se scrie un subprogram primește prin intermediul a doi parametri A și N , un vector și lungimea sa și care returnează lungimea maximă a unei secvențe de elemente crescătoare. *Exemplu* $N=8$ și $A=(5, 3, 3, 7, 7, 7, 2, 2)$ subprogramul va returna 5. Complexitate cerută: $O(N)$. **(10p)**
- Realizați un subprogram recursiv care afișează primii K factori primi ai unui numărul natural N ($N < 10^{17}$). Subprogramul va avea ca parametri numerele K , N și D (divizorul curent). Complexitate cerută: $O(\sqrt{N})$. **(10p)**
- Se consideră doi vectori $A(N)$ și $B(M)$ ce conțin valori naturale ordonate, $A(N)$ crescător și $B(M)$ descrescător. Să se scrie o secvență de instrucțiuni care afișează, în ordine crescătoare elementele care se regăsesc în ambii vectori, folosind operația de interclasare. Complexitate cerută $O(N+M)$ *Exemplu* $A=(2, 4, 5, 5, 7, 8)$ și $B=(16, 5, 4, 2, 2)$ se va afișa 2 4 5. **(10p)**
- Se dorește ca operația *Find-Replace* să fie executată asupra unui text care nu conține mai mult de 20000 de caractere. Această operație constă în înlocuirea tuturor aparițiilor unui subșir $s1$ cu un alt subșir $s2$. Scrieți un program care simulează această operație. Complexitate $O(N^2)$ **(20p)**
- Zăhărel* este un tip care se plictisește repede la școala. Într-o zi cu soare, când n-avea chef să asculte ce preda profesorul de matematică s-a apucat să deseneze puncte pe o foaie de matematică. El a desenat $N \leq 1.000$ astfel de puncte și apoi și-a pus următoarea întrebare: câte trapeze se pot forma cu vârfurile în aceste puncte? (doar era la ora de mate!). Un trapez este un patrulater convex cu cel puțin doua laturi paralele. Ajutați-l pe *Zăhărel* să determine câte trapeze poate forma cu cele N puncte de pe foaia de matematică, știind că oricare trei puncte sunt necoliniare. Pe prima linie din fișierul de intrare `trapez.in` se găsește numărul natural N . Pe următoarele N linii se găsesc perechi de numerele naturale reprezentând coordonatele punctelor, numere întregi din intervalul $[0, 2.000.000.000]$. Pe prima linie din fișierul de ieșire `trapez.out` se va găsi numărul de trapeze care se pot forma. Complexitate cerută: $O(N \log N)$ **(20p)**.

5	trapez.in		1	trapez.out
10 15				
3 6				
4 6				
4 2				
1 2				