

Testare finală - Grupa de excelență CNILC1 semigrupa 1 An școlar 2014-2015

Partea I (20p)

1. Se consideră doi vectori A și B cu N elemente valori naturale. Se dorește gruparea tuturor elementele lor în N perechi de forma $(a[i], b[j])$, astfel încât suma $a[i]+b[j]$ să fie aceeași pentru orice pereche. Afișați o posibilă grupare pentru $N=4$, $A=(4\ 2\ 1\ 5)$ și respectiv $B=(5\ 3\ 2\ 6)$ **(10p)**

2.

1 Algoritmul următor determină secvența de
2 produs maxim dintr-un vector cu numere întregi
3 nenule. Modificați algoritmul astfel încât să
4 furnizeze răspuns corect și dacă șirul conține
5 valori nule.
6
7

8
9
10 *Exemplu:* Pentru $N=10$ și șirul: (1 2 -6 3 4 5 -2
11 10 -5 6), algoritmul va afișa 3 4 5 -2 10 -5 6
12
13 **(10p)**
14
15
16
17
18
19
20

```
p1 ← 1; p2 ← 0; ic ← 1; sf ← 1;
pc1 ← 1; Max ← a[1];
pentru i ← 1, N executa
  p1 ← p1 * a[i]
  daca p2 ≠ 0 atunci
    p2 ← p2 * a[i]
  daca (a[i]<0) si (p2=0) atunci
    p2 ← 1; pc2 ← i + 1
  daca p1 > Max atunci
    max ← p1; ic ← pc1; sf ← i
  daca p2 > Max atunci
    max ← p2 ; ic ← pc2; sf ← i
pentru i ← ic, sf executa
  scrie a[i];
```

Partea II (70p)

1. Fie un vector $A(N)$ ce conține numere întregi. Să se scrie secvența de instrucțiuni care determină lungimea maximă a unei secvențe de elemente consecutive de valori egale. *Exemplu* $N=8$ și $A=(2, 3, 3, 7, 7, 7, 2, 2)$ se va afișa 3. Complexitate cerută: $O(N)$. **(10p)**
2. Realizați o secvență de instrucțiuni care determină toate numerele prime mai mici decât numărul natural N ($N < 1000000$), folosindu-se de algoritmul numit „Ciurul lui Eratostene”. Complexitate cerută: $O(N\sqrt{N})$. **(10p)**
3. Se consideră doi vectori $A(N)$ și $B(M)$ ce conțin valori naturale ordonate crescător. Să se scrie o secvență de instrucțiuni care afișează elementele pare distincte din doi vectori în ordine crescătoare, folosind operația de interclasare. Complexitate cerută $O(N+M)$ *Exemplu* $A=(2,4,5,6,7,8)$ și $B=(3, 4, 5, 16)$ se va afișa 2 4 6 8 16. **(10p)**
4. Să considerăm un șir S de caractere, care pot fi doar litere mici ale alfabetului englez. Numim secvență palindromică o succesiune de litere din șir care are proprietatea palindromică (fie că o parcurgem de la stânga la dreapta, fie că o parcurgem de la dreapta la stânga, secvența este aceeași). Scrieți secvența de instrucțiuni care determină cea mai lungă secvență palindromică dintr-un șir S dat. De exemplu pentru șirul **anaareuncojocasafrumos** se va afișa **cojoc**. Complexitate $O(N^2)$. **(20p)**
5. Să ne imaginăm o rețea formată din noduri situate în punctele de coordonate întregi, fiecare nod fiind unit prin bare paralele cu axele de coordonate de cele 4 noduri vecine. Un păianjen este plasat inițial în originea sistemului de coordonate. La fiecare secundă, păianjenul se poate deplasa din nodul în care se află în unul dintre cele 4 noduri vecine. Scrieți un program care să determine în câte moduri se poate deplasa păianjenul din poziția inițială, într-o poziție finală dată, în timpul cel mai scurt. Fișierul de intrare `spider.in` conține pe o singură linie abscisa și ordonata punctului final, separate prin spațiu: $x\ y$ ($0 < x, y \leq 80$) În fișierul de ieșire `spider.out` se va afișa pe prima linie numărul de moduri determinat N_r . Complexitate cerută: $O(X*Y)$ *Exemplu:* **(20p)**

2 3 `spider.in` | `spider.out` 10

Se acorda **10 p** din oficiu