**Analiza Algoritmilor - Test 1**

**Seria CA - 12.12.2015**

1. (0.8p) a) Se considera A = {multimea cuvintelor din dictionarul unei limbi}. Multimea A este numarabila sau nenumarabila?

b) Se considera B = {‘rosu’, ‘galben, ‘albastru’}. Multimea C = {cuvintele obtinute prin concatenarea de ori cate ori a elementelor multimii B} este numarabila sau nenumarabila? Justificati raspunsurile pe scurt.

2. (1.2p) Fie multimea A = {x | y, z, t, yzt, a.i. (x,y), (x,z), (x,t) B si y = z2}, unde B este recursiv numarabila. Demonstrati ca A este recursiv numarabila.

3. (1p) Fie f, g : -> + doua functii de complexitate. Aratati ca: O((f + g)2) = O(f2 + g2).

*Hint: se foloseste inegalitatea intre media aritmetica(ma) si media geometrica(mg), adica ma mg.*

4. (0.7p) Care este relatia cea mai potrivita pentru a exprima complexitatea secventei de cod:

*if (conditie)*

*instructiune1*

*else*

*instructiune2*

*5.* (2p) Rezolvati recurentele de complexitate prin ce metoda doriti:

a) T(n) = T(sqrt(n)) + (1)

b) T(n) = 2T(n/4) + sqrt(481) sqrt(n)(logn)7

c) T(n) = sqrt(n)T(sqrt(n)) + 5n

6. (2.8p) Determinati complexitatea, cat mai stransa, pentru secventele de cod:

|  |  |  |
| --- | --- | --- |
| for(int i=1; i<=n; i+=10)  for(int j=1; j<=n/2; j+=4)  ; | for(int i=1; i<=n ; i\*=2){  for(int j=1; j\*j<=n ; j++)  ;  for(int k=n; k>=2 ; k/=2)  ;  } | for(int i=1; i<=n ; i++)  for(int j=1; j<=i ; j++)  for(int k=j; k<=i+j ; k++)  for(int l=1; l<= i+j-k ; l++)  ; |

7. (1.5p) Se considera, ca date de intrare, un intreg k si trei vectori sortati A[1…n], B[1…n], C[1…n], de aceeasi dimensiune. Descrieti un algoritm eficient care calculeaza al k-lea cel mai mic element din reuniunea AB C.

8. (1p) Se considera o structura de date vazuta ca o lista de elemente cu operatiile urmatoare:

- qpush(x) : adauga un element x la capatul stanga al listei;

- qpop() : elimina elementul din lista situat cel mai la stanga;

- qpull() : elimina elementul din lista situat cel mai la dreapta;

Implementati aceasta structura de date folosind trei stive astfel incat costul amortizat pentru fiecare operatie qpush(x), qpop(), qpull() este (1). Fiecare element din structura de date considerata se gaseste, la orice moment de timp, in exact o singura stiva. Operatiile posibile pentru stiva sunt push(x) si pop. Se va folosi metoda potentialului pentru rezolvarea problemei.