

Subiectul III (30 de puncte)

Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Algoritmul de generare a tuturor numerelor de 3 cifre, toate nenule, este echivalent cu algoritmul de generare a: (6p.)
- a. submulțimilor unei mulțimi cu 3 elemente b. combinațiilor de 10 cifre luate câte 3
- c. aranjamentelor de 9 elemente luate câte 3 d. produsului cartezian a 3 mulțimi de cifre nenule

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Pentru subprogramul `suma` definit alături scrieți valoarea expresiei `suma(3,2)`. (4p.)
- ```
int suma (int a,int b)
{ if ((a==0) && (b==0)) return 0;
 if (a==0) return 1+suma(a,b-1);
 return 1+suma(a-1,b);
}
```

```
int suma (int a,int b)
{ if ((a==0) && (b==0)) return 0;
 if (a==0) return 1+suma(a,b-1);
 return 1+suma(a-1,b);
}
```
3. a) Scrieți definiția completă a subprogramului `shift` care primește prin intermediul parametrului `n` o valoare naturală nenulă ( $n \leq 100$ ), iar prin intermediul parametrului `x`, un tablou unidimensional cu maximum 100 de componente. Fiecare componentă a acestui tablou este un număr întreg care are cel mult 4 cifre. Subprogramul permută circular cu o poziție spre stânga, primele `n` elemente ale tabloului `x` și furnizează tabloul modificat tot prin parametrul `x`.  
**Exemplu:** dacă înainte de apel `n=4` și `x=(1,2,3,4)`, după apel `x=(2,3,4,1)`. (4p.)
- b) Scrieți un program C/C++ care citește de la tastatură o valoare naturală nenulă `n` ( $n \leq 100$ ), apoi cele `n` elemente ale unui tablou unidimensional `x`. Programul va inversa ordinea elementelor tabloului `x` folosind apeluri utile ale subprogramului `shift` și va afișa pe ecran, separate prin câte un spațiu, elementele tabloului rezultat în urma acestei prelucrări.  
**Exemplu:** dacă se citesc pentru `n` valoarea 5, iar tabloul `x` este `(1,2,3,4,5)` programul va determina ca `x` să devină `(5,4,3,2,1)`. (6p.)
4. Fișierul text `BAC.TXT` conține pe prima linie un număr natural nenul `n` ( $1 \leq n \leq 1000$ ), iar pe fiecare dintre următoarele `n` linii câte două numere întregi `a` și `b` ( $1 \leq a \leq b \leq 32000$ ), fiecare pereche reprezentând un interval închis de forma `[a,b]`. Scrieți un program C/C++ care citește numerele din fișier și determină un interval dintre cele citite care conține cel mai mare număr de numere întregi și afișează pe o linie a ecranului, separate printr-un spațiu, numerele care reprezintă capetele intervalului determinat. În cazul în care sunt mai multe intervale care îndeplinesc această proprietate, se vor afișa informațiile referitoare la acel interval la care numărul care reprezintă capătul din dreapta este minim.  
**Exemplu:** dacă fișierul `BAC.TXT` are conținutul alăturat, pe ecran se va afișa: 8 15 (pentru că numărul maxim de numere întregi aflate într-un interval este 8. Sunt două intervale care conțin 8 numere: `[17;24]` și `[8;15]`. Dintre acestea, `[8;15]` are capătul din dreapta cel mai mic). (10p.)
- ```
4
17 24
-2 3
9 15
8 15
```

```
4
17 24
-2 3
9 15
8 15
```