

Subiectul III (30 de puncte)

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

1. Ce valoare are $F(2758)$, pentru funcția F , definită alăturat? (4p.)

```
function F(x:integer):integer;  
begin  
    if x = 0 then F:= 0  
    else  
        if x mod 10 mod 2=0 then  
            F:=2+F(x div 10)  
        else F:= 10 - F(x div 10)  
    end;
```

- a. 0 b. 20 c. 12 d. 4

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

2. Variabilele i , j și aux sunt de tip întreg, iar elementele tabloului unidimensional x sunt următoarele: $x_1=10$, $x_2=5$, $x_3=-6$, $x_4=7$, $x_5=0$, $x_6=-2$.

Ce valori se vor afișa în urma executării secvenței de program alăturate? (6p.)

```
for i := 1 to 4 do  
    for j := i + 1 to 6 do  
        if x[i] > x[j] then  
            begin  
                aux := x[i];  
                x[i] := x[j];  
                x[j] := aux  
            end;  
    for i := 1 to 6 do  
        write(x[i], ' ');
```

3. a) Scrieți definiția completă a funcției **UltimaCifra** care primește prin cei doi parametri a și b câte un număr natural ($0 < a < 1000000$, $0 < b < 1000000$), calculează în mod eficient din punct de vedere al timpului de executare și returnează ultima cifră a numărului a^b (a la puterea b). (6p.)

b) Descrieți succint, în limbaj natural, metoda de rezolvare folosită, explicând în ce constă eficiența ei (3 – 4 rânduri) (4p.)

4. Fișierul text **SIR.IN** conține pe prima sa linie un număr natural n ($0 < n < 1001$), iar pe fiecare dintre următoarele n linii câte o pereche de numere naturale, x_i y_i ($1 \leq i \leq n$, $x_i \leq 30000$, $y_i \leq 30000$).

Scrieți programul **Pascal** care citește numerele din fișierul **SIR.IN** și scrie în fișierul text

SIR.OUT ultima cifră expresiei: $X_1^{y_1} + X_2^{y_2} + \dots + X_n^{y_n}$, folosind apeluri ale funcției **UltimaCifra**, definită la punctul 3.

Exemplu: dacă fișierul **SIR.IN** are conținutul alăturat, atunci **SIR.OUT** va conține numărul 2. (10p.)

```
3  
125 26  
458 146  
323 4589
```