

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

S-a notat cu $a \% b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întregă a numărului real c .

- Scrieți valoarea afișată dacă se citește numărul 27102. **(6p.)**
- Scrieți trei numere distincte din intervalul $[10, 9999]$ care pot fi citite, astfel încât în urma executării algoritmului, pentru fiecare dintre acestea, valoarea afișată să fie 11. **(6p.)**
- Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**
- Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind prima structură `cât timp...execută` cu o structură de tip `pentru...execută`. **(6p.)**

```

citește n
(număr natural)
m ← 0; x ← 1
cât timp x ≤ 9 execută
    cp ← n
    cât timp cp ≠ 0 execută
        dacă cp % 10 = x atunci
            m ← m * 10 + x
        cp ← [cp / 10]
    x ← x + 1
scrie m
    
```

- Fiind date două șiruri de caractere a și b , îl numim pe a **sufix** al lui b dacă a este egal cu b sau dacă b se poate obține din a prin alipirea la stânga a unor noi caractere. Variabilele a și b pot memora câte un șir cu cel mult 20 de caractere. Scrieți o secvență de instrucțiuni în urma executării căreia variabila a să memoreze un sufix al lui b format din trei caractere, sau șirul **nedeterminat**, dacă nu există un astfel de sufix.

Exemplu: dacă b memorează șirul centaur, atunci a memorează șirul aur, iar dacă b memorează șirul au, atunci a memorează șirul **nedeterminat**. **(6p.)**

- În declarațiile alăturate, variabila p memorează coordonatele unui punct în sistemul de coordonate xOy , iar variabila c memorează datele caracteristice ale unui cerc: lungimea razei și coordonatele centrului său. Scrieți o expresie care are valoarea 1 dacă și numai dacă punctul corespunzător variabilei p coincide cu centrul cercului corespunzător variabilei c . **(6p.)**

```

struct punct
{
    int x, y;
} p;
struct cerc
{
    struct punct centru;
    float raza;
} c;
    
```

SUBIECTUL al III-lea

(30 de puncte)

- Subprogramul **MaxImp** are doi parametri, a și b , prin care primește câte un număr natural ($2 \leq a < b \leq 400$). Subprogramul returnează cel mai mare număr natural din intervalul $[a, b]$ pentru care produsul divizorilor săi impari pozitivi este strict mai mare decât el însuși sau 0, dacă nu există niciun astfel de număr. Scrieți definiția completă a subprogramului.

Exemplu: dacă $a=14$ și $b=19$, atunci subprogramul returnează 18 ($1 \cdot 3 \cdot 9 = 27 > 18$). **(10p.)**

- Numim **pătrat** de dimensiune m al unui tablou bidimensional tabloul obținut din acesta păstrând doar elementele aflate pe primele m linii și pe primele m coloane ale sale.

Scrieți un program C/C++ care citește de la tastatură un număr natural, n ($n \in [2, 20]$), apoi elementele unui tablou bidimensional cu n linii și n coloane, numere naturale din intervalul $[0, 10^4]$. Programul determină un pătrat de dimensiune maximă al tabloului citit, cu toate elementele egale, și afișează pe ecran valoarea acestei dimensiuni.

Exemplu: pentru $n=5$ și tabloul alăturat, se afișează pe ecran 3. **(10p.)**

```

2 2 2 2 2
2 2 2 2 8
2 2 2 2 2
2 1 2 8 7
3 5 2 1 2
    
```

- Fișierul **bac.txt** conține un șir de cel puțin două și cel mult 10^6 numere naturale din intervalul $[0, 10^3]$, separate prin câte un spațiu. Șirul are cel puțin un termen par și cel puțin un termen impar. Se cere să se afișeze pe ecran termenii șirului, separați prin câte un spațiu, astfel încât toți cei impari să apară înaintea tuturor celor pari, și atât subșirul format din cei impari, cât și subșirul format din cei pari, să fie în ordine crescătoare, ca în exemplu. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.

Exemplu: dacă fișierul conține numerele 12 2 3 1 2 5

se afișează pe ecran: 1 3 5 2 2 12

- Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
- Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**