



2. Algoritmul alăturat este reprezentat în pseudocod. S-a notat cu $a \div b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .
- Scrieți ce se va afișa dacă se citește valoarea 2019.
 - Pentru câte numere întregi de exact 3 cifre algoritmul va afișa mesajul DA, urmat de o valoare numerică?
 - Scrieți programul C/C++ corespunzător algoritmului dat.
 - Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască structurile `cât_timp` ... `execută` cu structuri repetitive cu test final.
- ```

citește x (număr natural nenul)
ok←-1, y←x
cât timp x>9 execută
 dacă ([x/10]%10>=x%10) atunci
 ok←0
 x←[x/10]
dacă ok=0
 atunci scrie "NU"
 altfel
 x←y, y←x%10, x←[x/10], p←-1
 cât timp x>0 execută
 c←x%10, y←y+c*p
 y←y*10+c, p←p*100, x←[x/10]
 scrie "DA ",y

```
3. Variabilele  $i$  și  $j$  sunt de tip întreg, iar variabila  $a$  memorează un tablou unidimensional cu 36 de elemente, numerotate de la 0 la 35, având inițial toate elementele egale 0. Fără a utiliza alte variabile, scrieți secvența de instrucțiuni de mai jos, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila  $a$  să memoreze șirul de numere alăturat.
- ```

for (i=0; i<36; i++)
    .....
    
```
- | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 2 | 3 | 5 | 8 | 3 | 1 | 4 | 5 | 9 | 4 | 3 | 7 | 0 | 7 | 7 | 4 | 1 | 5 | 6 | 1 | 7 | 8 |
| 5 | 3 | 8 | 1 | 9 | 0 | 9 | 9 | 8 | 7 | 5 | 2 | | | | | | | | | | | | |
- (6p.)

SUBIECTUL al III-lea

(30 de puncte)

- Se citesc două numere naturale a și b , de maximum 4 cifre fiecare. Se cere să se determine în variabilele c și d , cel mai mic, respectiv cel mai mare număr din intervalul $[a, b]$, care e format doar din cifre distincte sau valoarea -1, dacă aceste numere nu există. Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate. **(10p.)**
- CodCrip** este o nouă metodă de criptare a textelor în era digitală. Elevii noștri doresc să îi ajutați să-și codifice mesajele folosind metoda CodCrip: *oricare două coduri cu aceeași paritate aflate pe poziții consecutive să apară în ordine inversă. Se garantează că nu vor exista mai mult de două coduri consecutive cu aceeași paritate.* Scrieți un program C/C++ care citește un număr natural n ($n \leq 250$), urmat de n valori numerice reprezentând codurile ASCII ale caracterelor din mesaj, separate prin câte un spațiu și afișează mesajul criptat. Exemplu: $n=13$ și codurile: 98 97 99 108 97 117 114 101 97 116 117 108 Se vor afișa modificările: 98 99 97 108 117 97 114 97 101 116 117 108 **(10p.)**
- Fișierul text `bac.in` conține pe primul rând un număr natural nenul n , $n \leq 1000$, pe al doilea rând un șir de n numere întregi ordonate descrescător, iar pe rândul următor cel mult 1000 de numere întregi distincte. Valorile din cele două șiruri sunt numere întregi cu maximum 9 cifre fiecare. Folosind un algoritm eficient din punctul de vedere al spațiului de memorie și al timpului de executare, scrieți un program C/C++ care citește numerele din fișier și determină media aritmetică a numerelor de pe ultimul rând din fișier care se regăsesc printre cele n numere de pe al doilea rând sau mesajul **NU EXISTA**, în cazul în care nu există astfel de numere.
 - Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
 - Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)****Exemplu:** dacă fișierul `bac.in` are următorul conținut:

```

5
10 9 8 7 6
1 9 6 23 11 12 32, pe ecran se va afișa 7.50, adică (9+6)/2.
    
```