

Setul 2 - Programe simple ce folosesc instrucțiunile de bază ale limbajului C

Se recomandă ca în timpul orelor de laborator să se rezolve problemele **1, 3, 6, 9, 10, 11, 13**. Problemele marcate cu * au un grad de dificultate ceva mai ridicat pt. incepatori.

1. Scrieți un program care testează dacă o literă introdusă de la tastatură este literă mare, literă mică, cifră sau alt caracter.
2. * Scrieți un program care citește de la tastatură 3 numere întregi, evaluează dacă ele pot reprezenta laturile unui triunghi și, dacă da, verifică despre ce tip de triunghi este vorba (echilateral, dreptunghic, isoscel, oarecare) și afișează informația pe ecran.
3. * Scrieți un program care testează dacă un număr natural introdus de la tastatură este prim.
4. Scrieți un program ce găsește și afișează toate numerele naturale pare mai mici ca **n** (**n** este introdus de la tastatură).
5. Scrieți un program care să afișeze primii **n** (**n** citit de la tastatură) termeni ai șirului Fibonacci știind că șirul lui Fibonacci se construiește după următoarele reguli:

$$F_0 = 0$$

$$F_1 = 1$$

$$F_k = F_{k-1} + F_{k-2} \quad \text{pentru orice } k > 1$$

Ca variantă, afișați termenii șirului lui Fibonacci ce au valoare mai mică decât **v**, citit de la tastatură.

*** Dacă vreți să aflați mai multe despre șirul lui Fibonacci și relația sa cu viața care ne înconjoară, un punct de plecare poate fi <http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fibnat.html>

6. Citiți de la tastatură o secvență de valori numerice până la întâlnirea valorii 0. Calculați și afișați câte valori negative, respectiv pozitive au fost citite. Scrieți variante de program pentru valori numerice întregi, respectiv reale. Atenție la compararea cu zero a valorilor de tip real.
7. Folosiți instrucțiunea **do...while** pentru a forța utilizatorul să introducă date corecte (conforme cu diverse cerințe). De exemplu: un număr natural nu poate fi negativ. (Exersarea exemplului de program prezentat la curs).
8. Aceeași cerință ca în cazul problemei nr. 7 pentru a solicita utilizatorului să introducă opțiunea de a continua sau de a încheia un program ("Doriți să continuați? D/N").
9. Pentru **n** citit de la tastatură, să se calculeze și să se afișeze suma: **S = 1 + 1*2 + 1*2*3 + ... n!**
10. Să se determine și să se afișeze cifrele unui număr natural **n** citit de la tastatură.
11. Afișați, tabelat, tabla înmulțirii cu numere de la 1 la 10. De exemplu:

1x1=1	1x2=2	...	1x10=10
.....			
10x1=10	10x2=20	...	10x10=100

12. Citiți de la tastatură un număr natural **n**. Să se afișeze **n** pătrate, ca în exemplul următor (unde **n** are valoarea 3):

```
1
2 2
2 2
3 3 3
3 3 3
3 3 3
```

13. Să se scrie un program care să citească un număr întreg, iar dacă acest număr este 1 să afișeze *luni*, dacă este 2 să afișeze *marți*, ... dacă este 7 să afișeze *duminică*, iar dacă nu este cuprins între 1 și 7 să afișeze cuvântul *eroare*. Să se scrie câte o variantă de program cu instrucțiunea **switch**, respectiv **if**.
14. Folosiți instrucțiunea **for** pentru a citi de la tastatură **n** valori întregi și a calcula media lor aritmetică. Scrieți apoi variantele de program ce folosesc instrucțiunea **while**, respectiv **do...while** și observați deosebiri/asemănări.