

1. Având o tablă de șah și un rege așezat pe o poziție random pe acea tablă, să se parcurgă întreaga tablă de șah, astfel încât regele să treacă cel mult o dată printr-o căsuță.

2. Având o tablă de șah și un cal așezat pe o poziție random pe acea tablă, să se parcurgă întreaga tablă de șah, astfel încât calul să treacă cel mult o dată printr-o căsuță.

3. Se citește un număr n . Generați și afișați toate combinațiile de câte n cifre binare care nu au două cifre de 1 alăturate.

4. Se citesc n numere naturale. Determinați o așezare a acestor numere sub forma unui cerc, astfel încât suma produselor de câte două numere alăturate să fie maximă.

date.in

6

1 8 3 2 5 4

date.out

1 2 4 8 5 3

5. Se citește un număr natural n . Afișați permutările mulțimii $1, 2, 3, \dots, n$ în care până la jumătate elementele sunt în ordine descrescătoare, iar de la jumătate până la final în ordine crescătoare.

6. Se citește un număr natural n . Afișați permutările mulțimii $1, 2, 3, \dots, n$ în care până la jumătate elementele sunt în ordine crescătoare, iar de la jumătate până la final în ordine descrescătoare.

7. Se citesc două numere naturale n și k , k fiind mai mic decât numărul de cifre al numărului n . Afișați cel mai mare număr care se poate forma eliminând k cifre din numărul n .

$n=3452234$

$k=4$

numarul cautat este 544

8. Se citește un număr natural n . Afișați permutările mulțimii $1, 2, 3, \dots, n$ în care elementele pare sunt puncte fixe (se află pe poziție egale cu valoarea lor).

9. Se citește un număr natural n . Afișați permutările mulțimii $1, 2, 3, \dots, n$ în care elementele impare sunt puncte fixe (se află pe poziție egale cu valoarea lor).

10. Folosind metoda divide et impera, scrieți o funcție care să determine dacă un tablou cu elemente întregi este ordonat crescător.

11. Se citesc 3 numere naturale S , n și e cu următoarele semnificații: S este o sumă de bani care trebuie plătită folosind bancnote care au valori puterile lui e de la 0 la n . Să se afișeze modalitatea de plată folosind un număr minim de bancnote de tipurile precizate. Să se afișeze la final numărul de bancnote folosite. (256, 10, 2)

12. Într-o sală de spectacol trebuie planificate n spectacole într-o zi. Pentru fiecare spectacol se cunosc ora de început și ora de terminare (numere întregi). Să se planifice un număr maxim de spectacole astfel încât să nu fie două spectacole care să se suprapună.

```
date.in
7
2 4
8 11
5 6
5 8
3 7
7 8
9 12
date.out:
2,4 5,6 7,8 9,12
```

13. Sa se verifice daca intr-o expresie se parantezele se inchid corect (a{b[c]d}e // corect //a[b{c}d]e} incorect).

14. O padure este împărțită în n X m zone. In fiecare zona creste cate un pom. Din fiecare pom cade pe jos o cantitate de ghinde. In zona stânga sus se află o veveriță care vrea să ajungă în zona dreapta jos. Veverița se poate deplasa doar pe două direcții: în jos sau spre dreapta.

Determinați cantitatea maximă de fructe pe care le poate aduna veverița prin deplasarea din poziția inițială în cea dorită.

3 3 ----- output 7

0 4 1

0 1 1

1 0 1

15. Creați două liste: o listă ce conține numere pare și o listă ce conține numere impare. Numerele se vor citi de la tastatură până când se introduce -1.

16. Creați o listă ce conține un anumit număr de numere reale. Scrieți o funcție ce introduce un nod ce conține media fiecăror două noduri vecine, între cele două noduri.

17. Să se construiască o listă liniară simplu înălțuită cu elemente numere întregi. Să se inverseze sensul legăturilor din listă.

18. Să se construiască o listă liniară simplu înălțuită cu elemente numere întregi. Să se șteargă toate numerele pare.

19. Creați o listă ce conține un anumit număr de numere reale. Scrieți o funcție ce introduce un nod ce conține diferența fiecăror două noduri vecine, între cele două noduri.

20. După o vacanță de 3 luni, te întorci acasă și găsești o gramadă de scrisori în cutia poștală. Având în vedere că printre scrisori se află și facturi ce trebuie plătite urgent și diverse reclame, trebuie să le sortezi în funcție de prioritate. Elaborează un astfel de algoritm în C sau C++ pentru a transformă stiva de corespondență într-o coadă de priorități.

21. Implementati un algoritm de cautare binara, care imparte sirul in trei parti aproximativ egale, nu in doua.

22. Avem doua array-uri sortate A si B de marime n. Scrieti un algoritm care gaseste mediana arrayului obtinut dupa ce doua arrayuri au fost unite (lungimea arrayului este 2n).

23. Dat fiind un array de task-uri, in care fiecare task are un deadline si un profit asociat daca task-ul este finalizat inainte de deadline. Fiecare task se termina intr-o unitate de timp, deci cel mai scurt deadline este 1. Sa se aranjeze task-urile astfel incat profitul sa fie maximizat. (GREEDY)

JobID	Deadline	Profit
a	2	100
b	1	19
c	2	27
d	1	25
e	3	15

Output:

c, a, e

24. Se da o tija de n cm si un array de preturi care contine preturile tuturor bucatilor de tija de marime mai mica decat n. Sa se determine valoarea maxima ce se poate obtine daca se taie tija si se vinde pe bucati. (Ex. daca lungimea este 8 si valorile bucatilor sunt urmatoarele atunci profitul maxim este 22). (programare dinamica)

```
lungime | 1 2 3 4 5 6 7 8
-----
pret    | 1 5 8 9 10 17 17 20
```

25. Să se scrie un program care afiseaza toate permutarile unui string. (ABC - ABC, ACB, BAC, BCA, CAB, CBA). Folositi un algoritm de backtracking.

26. Dat fiind un array sortat si un numar x, sa se gaseasca o pereche in array astfel incat suma elementelor perechii sa fie cea mai apropiata de x.

Input: arr[] = {10, 22, 28, 29, 30, 40}, x = 54

Output: 22 and 30

27. Calculați mulțimea numerelor complexe c Mandelbrot, definite prin: $f(n+1) = f(n)^2 + c$, $f(0) = c$

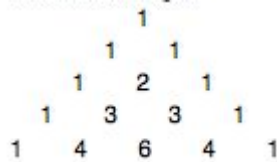
28. Scrieți un program în C care să conțină o funcție recursivă și una iterativă care preia un parametru de tip string și un parametru ce reprezintă lungimea stringului și care returnează același string, dar în inversat.

29. Scrieți un program în C care să conțină o funcție recursivă și o funcție iterativă `count_spaces(char *s)` care numără câte spații libere apar într-un string.

30. Scrieți un program în C care să conțină o funcție recursivă și o funcție iterativă `replace(char *s, char from, char to)` care schimbă toate aparițiile lui `from` din `s` în `to`. (`replace(steve, e,a) = stava`).

31. Scrieți un program în C care să afișeze o listă simplu înlănțuită în ordine inversă

32. Scrieți un program în C care să afișeze triunghiul lui Pascal pentru `n` dat.



Triunghiul lui Pascal pentru n=5:

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
```

33. Implementați MergeSort pentru liste simplu înlănțuite.

34. Implementați QuickSort pentru liste simplu înlănțuite.

35. Implementați InsertionSort pentru liste simplu înlănțuite.

36. Implementați ShellSort pentru liste simplu înlănțuite.

37. Implementați SelectionSort pentru liste simplu înlănțuite

38. Verificati daca un numar este palindrom

39. Verificati daca un numar este par sau impar.

40. Scrieti un program care afla maximul din 4 numere.

41. Scrieti un program care interschimba 3 valori. (input `a = 2, b=3, c=4`,
output:`a=3, b=4, c=2`)

42. Scrieti un program care returneaza un numar inversat (ex. input `1234567`,
ouptu `7654321`)

43. Verificati daca un numar poate fi scris ca suma a doua numere prime

44. Scrieti un program care calculeaza suma primelor n numere naturale recursiv.

44. Scrieti un program care calculeaza produsul primelor n numere naturale recursiv.

45. Scrieti un program care rescrie invers o propozitie folosind recursivitate.
('afara este foarte cald', dlac etraof etse arafa)