

Laborator 1 - Recursivitate

- 1) Sa se implementeze o functie recursiva ce primeste doua numere intregi a si b si returneaza a la puterea b.

Exemplu:

a = 2, b = 3

r = 8

- 2) Sa se implementeze o functie recursiva ce va primi un numar x si o putere n si va calcula pentru x suma puterilor de la 0 la n.

Exemplu:

x = 2, n = 3

$r = 2^0 + 2^1 + 2^2 + 2^3 = 15$

- 3) Sa se implementeze o functie recursiva ce va calcula cel mai mare divisor comun a doua numere a si b aplicand algoritmul lui Euclid.

Varianta Iterativa:

```
function euclid(a, b)
    if a = 0
        return b
    while b ≠ 0
        if a > b
            a := a - b
        else
            b := b - a
    return a
```

Exemplu:

a = 12, b = 8

r = 4

- 4) Sa se scrie un program ce va citi de la tastatura 3 numere n, a si b, apoi n valori ce vor fi puse intr-un vector v. Apoi sa se implementeze o functie recursiva ce va primi vectorul si cele doua numere a si b si va returna suma elementelor de pe pozitiile dintre a si b, inclusiv.

Exemplu:

$n = 7, a = 2, b = 5, v = \{ 1, 2, -3, 0, 2, 1, 10 \}$

$r = 2 + -3 + 0 + 2 = 1$

- 5) Sa se scrie un program ce va citi de la tastatura doua numere a si b, cu $a < b$ si va ghici printr-un numar optim de incercari un numar din intervalul $[a, b]$ la care se gandeste utilizatorul.

Programul va folosi o functie recursiva ce va intreba utilizatorul o anumita valoare din intervalul dat si va citi de la tastatura 0, 1 sau -1, reprezentand :

0 = numarul a fost ghicit

1 = numarul ce trebuie ghicit e mai mare decat cel spus de program

-1 = numarul ce trebuie ghicit e mai mic decat cel spus de program

Indicatie: Cautare binara (intreaba intotdeauna mijlocul intervalului)

Exemplu:

$a = 5, b = 15$

(Cautam pe 8)

C : 10?

U: -1

C: 7?

U: 1

C: 8?

U: 0