



# *“Divide et Impera”*



**PREZENTAREA METODEI  
ȘI APLICAȚII**



## Descrierea metodei



- **Metoda Divide Et Impera** constă în împărțirea problemei de rezolvat în două sau mai multe probleme similare celei inițiale, dar de dimensiune mai mică și apoi combinarea soluțiilor pentru a crea o soluție a problemei inițiale.
- Procedurul se reia pentru fiecare din subproblemele obținute până când (în urma descompunerilor repetate) se ajunge la probleme ce admit rezolvare imediată.
- **OBS**: Deoarece problemele rezultate sunt similare celei inițiale, metoda se poate exprima recursiv, dar admite și varianta iterativă.



## *Etapele metodei*



- 1. Divide:** Se împarte problema în subprobleme de același tip, dar de dimensiune mai mică;
  - 2. Impera:** Se rezolvă fiecare dintre subprobleme – direct dacă acestea sunt simple – sau continuă cu **divide** prin reducerea acestora la alte subprobleme, recursiv;
  - 3. Impera:** Se combină soluțiile subproblemelor, pentru obținerea soluției problemei inițiale.
- **Obs:** Procesul de descompunere în subprobleme se oprește când acestea permit o rezolvare directă. Această metodă se aplică în general, pentru prelucrarea vectorilor dar și a altor tipuri de date.



# Aplicații



***Să se determine cea mai mare valoare dintr-un șir de  $n$  numere întregi, folosind metoda Divide et Impera.***

*Rezolvare:*

- Dacă șirul are un singur element, acesta va fi elementul maxim. Pentru un subșir oarecare de cel mult 2 elemente vom avea următoarele etape:
- Împărțim șirul inițial  $x [ p \dots q ]$  în două subșiruri  $x [ p \dots m ]$  și  $x [ m+1 \dots q ]$ , unde  $m$  este mijlocul șirului:  $m = \lfloor (p+q)/2 \rfloor$ . Cele două subșiruri pot fi împărțite la rândul lor în alte două șiruri până se ajunge la un subșir de dimensiune 1. Notăm cu  $x [ p \dots q ]$  subșirul format din toate elementele șirului dintre  $x[p]$  și  $x[q]$ .
- Se determină recursiv elementul maxim pentru cele două subșiruri (a și b).
- Se combină cele două maxime obținute pentru aflarea maximului din șirul inițial.



# Exemplu numeric



5	12	15	7	14	23	9	15
---	----	----	---	----	----	---	----

$s_1$

$s_2$

5	12	15	7	14	23	9	15
---	----	----	---	----	----	---	----

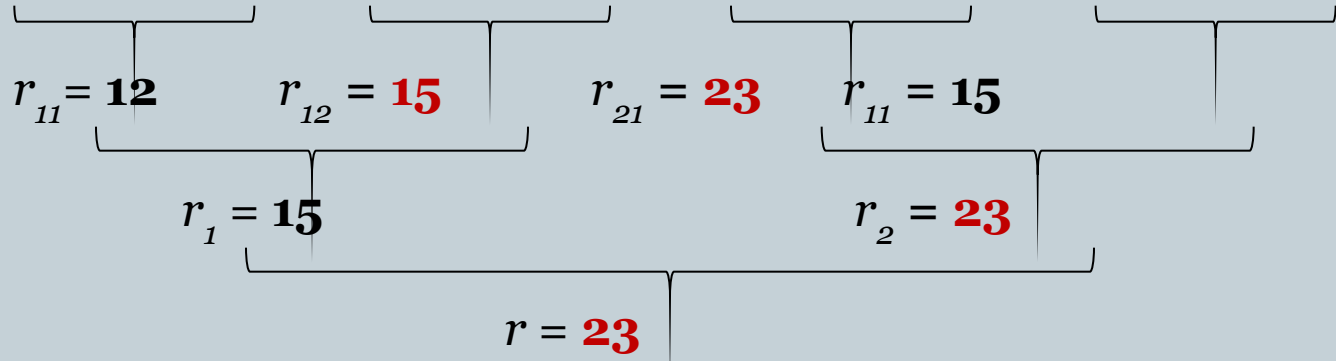
$s_{11}$

$s_{12}$

$s_{21}$

$s_{22}$

5	12	15	7	14	23	9	15
---	----	----	---	----	----	---	----





## *Subprogramul maxim*



```
Type vector=array[1..100] of integer;
Var x:vector;
    i,n:integer;

function maxim ( p , q : integer ) : integer;
var m, a, b : integer;
begin
    if p < q then begin
        m := ( p + q ) div 2;
        a := maxim ( p , m );
        b := maxim ( m + 1 , q );
        if a > b then maxim := a
            else maxim := b;
        end
    else maxim := x [ p ];
end;
```



## *Programul principal*



```
begin
  write('n='); readln(n);
  for i:=1 to n do
    begin
      write('x[', i, ']=');
      readln ( x[i] );
    end;
  writeln (' maximul=', maxim ( 1, n ));
  readln;
end.
```



## Aplicații grilă

### 1. Ce va afișa programul următor?

```
var v : array [ 1 .. 50 ] of integer ; i : integer;
```

```
function s ( a , b : byte ): longint;
```

```
begin
```

```
  if a > b then s := 0
```

```
    else if a=b then s := v [ a ]
```

```
    else s := s ( a , ( a + b ) div 2 ) + s ( ( a + b ) div 2 + 1 , b );
```

```
end;
```

```
begin
```

```
  for i := 1 to 20 do v [ i ] := i;
```

```
  writeln ( s ( 5 , 9 ) );
```

```
  readln;
```

```
end.
```

a) 29    b) 35    c) 45    d) 14





## Aplicații grilă



### 2. Ce va afișa programul pentru $n = 10$ ?

```
var n : integer;
```

```
function s ( a , b : integer ) : longint ;
```

```
var m : byte;
```

```
begin
```

```
    if a <= b then
```

```
        begin
```

```
            m := ( a + b ) div 2;
```

```
            s := m + s ( a , m-1 ) + s ( m+1 , b );
```

```
        end
```

```
        else s := 0;
```

```
    end;
```

```
begin readln(n);
```

```
    writeln ( s ( 1 , n ) );
```

```
end.
```

a) 29    b) 35    c) 41    d) 45



## Probleme propuse



1. Se citește  $n$  un număr natural. Să se calculeze **produsul primelor  $n$  numere naturale**  $P=1*2*...*n$ , folosind metoda Divide et Impera.
2. Se dau cele  $n$  elemente ale unui vector. Să se determine cu metoda Divide et Impera **suma elementelor din vector**.
3. Se citesc cele  $n$  elemente ale unui vector cu valori întregi. Să se determine **maximul dintre elementele impare din vector**, cu metoda Divide et Impera.