

## MERGE SORT (SORTAREA PRIN INTERCLASARE)

Se consideră un șir cu  $n$  elemente aleatoare. Să se ordoneze elementele șirului crescător.

Metoda, folosește algoritmul divide et impera și are la baza interclasarea a doi vectori: fiind dați doi vectori ordonați se obține un al treilea vector ordonat care va conține elementele din cei doi vectori

```
#include <iostream>
using namespace std;
int a[1000],n;
void interclas(int i,int m,int j)
{
    int b[1000];
    int x=i;
    int k=1;
    int y=m+1;
    while(x<=m && y<=j) //se sorteaza vectorul prin comparare element cu element
        if (a[x]<a[y])
            b[k++]=a[x++];
        else
            b[k++]=a[y++];
    while(x<=m)// ultimul element se pune in vectorul b
        b[k++]=a[x++];
    while (y<=j)// ultimul element se pune in vectorul b
        b[k++]=a[y++];
    int t=i; /*in t se memoreaza pozitia din care incep sa pun elemente
             in vectorul a*/
    for (k=1;k<=(j-i)+1;k++) //pun elemente in vectorul a
        a[t++]=b[k];
}
void divimp(int i,int j)
{
    if (i<j)
    {
        int m=(i+j)/2;
        divimp(i,m);
        divimp(m+1,j);
        interclas(i,m,j); //primele valori vor fi 1,1,2
    }
}
```

```

}
int main()
{
    cout<<"n=";
    cin>>n;
    for(int i=1;i<=n;i++)
    {
        cout<<"a["<<i<<"]=";
        cin>>a[i];
    }
    divimp(1,n);
    for(int i=1;i<=n;i++)
    cout<<a[i]<<' ';
}

```

Exemplu: Pentru sirul initial de mai jos se fac următoarele interclasari:

- Interclasare(1,1,2)
- Interclasare(3,3,4)
- Interclasare(1,2,4)
- Interclasare(5,5,6)
- Interclasare(5,6,7)
- Interclasare(1,4,7)

