**Merge sort**

Merge sort merupakan algoritma pengurutan yang dirancang untuk memenuhi kebutuhan pengurutan atas suatu rangkaian data yang tidak memungkinkan untuk ditampung dalam memori komputer karena jumlahnya yang terlalu besar.

Merge sort menggunakan pola divide and conquer. Dengan hal ini deskripsi dari algoritma dirumuskan dalam 3 langkah berpola divide-and-conquer. Berikut menjelaskan langkah kerja dari Merge sort :

1.       Divide

  Memilah elemen – elemen dari rangkaian data menjadi dua bagian.

2.         Conquer

Conquer setiap bagian dengan memanggil prosedur merge sort secara      rekursif.

  3.   Kombinasi

Mengkombinasikan dua bagian tersebut secara rekursif untuk mendapatkan    rangkaian data berurutan.

Proses rekursi berhenti jika mencapai elemen dasar. Hal ini terjadi bilamana bagian yang akan diurutkan menyisakan tepat satu elemen. Sisa pengurutan satu elemen tersebut menandakan bahwa bagian tersebut telah terurut sesuai rangkaian.

Contoh:

Inputan datanya adalah sebagai berikut:

|  |  |  |  |
| --- | --- | --- | --- |
| 7 | 2 | 6 | 5 |

Membagi rangkaian menjadi dua bagian:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 7 | 2 |  | 6 | 5 |

Kedua bagian tersebut bisa dinamai LeftArr dan RightArr

Membagi LeftArr menjadi dua bagian:

|  |  |  |
| --- | --- | --- |
| 7 |  | 2 |

Membandingkannya kemudian dikombinasikan:

|  |  |
| --- | --- |
| 2 | 7 |

7 dan 2 kemudian tukar posisi karena 2 < 7

Membagi RightArr menjadi dua bagian:

|  |  |  |
| --- | --- | --- |
| 6 |  | 5 |

Membandingkannya kemudian dikombinasikan:

|  |  |
| --- | --- |
| 5 | 6 |

6 dan 5 kemudian tukar posisi karena  5 < 6

Membandingkan LeftArr dan RightArr.

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | 7 | 5 | 6 |

Kemudian dilakukan proses membandingkan lagi antara angka di LeftArr dengan RightArr.

Pembandingan biasanya dimulai dari angka terdepan di masing-masing bagian.

Mengkombinasikan LeftArr dengan RightArr.

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | 5 | 6 | 7 |

Sehingga menjadi: 2<5<6<7

**Berikut ini contoh program dalam pascal**

program MergeSort;  
uses crt;  
type arr = array [1..100] of integer;  
  
var  
ArrMain,ArrUrut : arr;  
n,m : integer;  
function merge(Left:arr; pjgL:integer; Right : arr; pjgR:integer) :arr;  
var  
i,j,k,m,panjang: integer;  
hasil : arr;  
begin  
i:=1;  
j:=1;  
k:=1;  
panjang:=pjgL+pjgR;  
while ((pjgL>0) and (pjgR>0)) do  
begin  
if(Left[i]<= Right[j]) then  
begin  
hasil[k]:=Left[i];  
i:=i+1;  
k:=k+1;  
pjgL:=pjgL-1;  
end  
else  
begin  
hasil[k]:=Right[j];  
j:=j+1;  
k:=k+1;  
pjgR:=pjgR-1;  
end;  
end;  
while (pjgL>0) do  
begin  
hasil[k]:=Left[i];  
i:=i+1;  
k:=k+1;  
pjgL:=pjgL-1;  
end;  
while (pjgR>0) do  
begin  
hasil[k]:=Right[j];  
j:=j+1;  
k:=k+1;  
pjgR:=pjgR-1;  
end;  
merge:=hasil;  
for m:= 1 to panjang do  
writeln('Array Hasil ke-',m,' : ',hasil[m]);  
end;  
  
function mergesort(pjg:integer;A : arr):arr;  
var  
middle,i,pjgLeft,pjgRight : integer;  
ArrLeft,ArrRight,ArrHasil : arr;  
begin  
if pjg <= 1 then  
mergesort := A  
else  
begin  
middle := pjg div 2;  
for i:=1 to middle do  
ArrLeft[i]:=A[i];  
  
for i:=(middle+1) to pjg do  
ArrRight[i-middle]:=A[i];  
pjgLeft := pjg div 2;  
pjgRight := (pjg+1) div 2;  
for m:= 1 to pjgLeft do  
writeln('ArrayLeft ke-',m,' : ',ArrLeft[m]);  
for m:= 1 to pjgRight do  
writeln('ArrayRight ke-',m,' : ',ArrRight[m]);  
ArrLeft:=mergesort(pjgLeft,ArrLeft);  
ArrRight:=mergesort(pjgRight,ArrRight);  
mergesort:=merge(ArrLeft,pjgLeft,ArrRight,pjgRight);  
end;  
end;  
  
begin  
clrscr;  
write('Jumlah array : ');readln(n);  
for m := 1 to n do  
begin  
write('Array ke-',m,' : ');  
readln(ArrMain[m]);  
end;  
writeln;  
ArrUrut := mergesort(n,ArrMain);  
  
for m:= 1 to n do  
writeln('Array Urut ke-',m,' : ',ArrUrut[m]);  
end.