**MODUL PRAKTIKUM 4**

**ANALISIS ALGORITMA**



**Disusun oleh :**

Rifaa’ Zalfaa’ Fakhriyyah 140810170031

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS PADJADJARAN**

**2019**

1. Merge Sort

Source code:

/\*\*\*\*

Nama Program : mengurutkan elemen menggunakan merge sort

Nama        : Rifaa' Zalfaa' Fakhriyyah

NPM         : 140810170031

\*\*\*/

#include <chrono>

#include<iostream>

using namespace std;

using namespace std::chrono;

void Merge(int A[],int p, int q,int r) /\*It merges two arrays \*/

{

    int n1=q-p+1,i,j,k; /\*n1 is the size of L[]\*/

    int n2=r-q; /\*n2 is the sixe of R[]\*/

    int L[n1],R[n2];

    for(i=0;i<n1;i++)

    {

        L[i]=A[p+i];

    }

    for(j=0;j<n2;j++)

    {

        R[j]=A[q+j+1];

    }

    i=0,j=0;

    for(k=p;i<n1&&j<n2;k++)

    {

        if(L[i]<R[j])

        {

            A[k]=L[i++];

        }

        else

        {

            A[k]=R[j++];

        }

    }

    while(i<n1) /\*If L[] has more elements than R[] then it will put all the reamining elements of L[] into A[]\*/

    {

        A[k++]=L[i++];

    }

    while(j<n2) /\*If R[] has more elements than L[] then it will put all the reamining elements of R[] into A[]\*/

    {

        A[k++]=R[j++];

    }

}

void MergeSort(int A[],int p,int r) /\*It will will divide the array and sort them while merging them\*/

{

    int q;

    if(p<r)

    {

        q=(p+r)/2; /\*q is the middle element to divide the array\*/

        MergeSort(A,p,q);

        MergeSort(A,q+1,r);

        Merge(A,p,q,r);

    }

}

int main()

{

    int A\_Size; /\*A\_Size size of A[]\*/

    cout<<"Enter number of elements :";

    cin>>A\_Size;

    int A[A\_Size];

    cout<<"Enter the array elements :";

    for(int i=0;i<A\_Size;i++)

    {

        cin>>A[i];

    }

    // Use auto keyword to avoid typing long

// type definitions to get the timepoint

// at this instant use function now()

auto start = high\_resolution\_clock::now();

    MergeSort(A,0,A\_Size-1);

    // After function call

auto stop = high\_resolution\_clock::now();

// Get duration. Substart timepoints to

// get durarion. To cast it to proper unit

// use duration cast method

auto duration = duration\_cast<microseconds>(stop - start);

cout << "Time taken by function: " << duration.count() << " microseconds" << endl;

    cout<<"The array elements after merge sort :";

    for(int i=0;i<A\_Size;i++)

    {

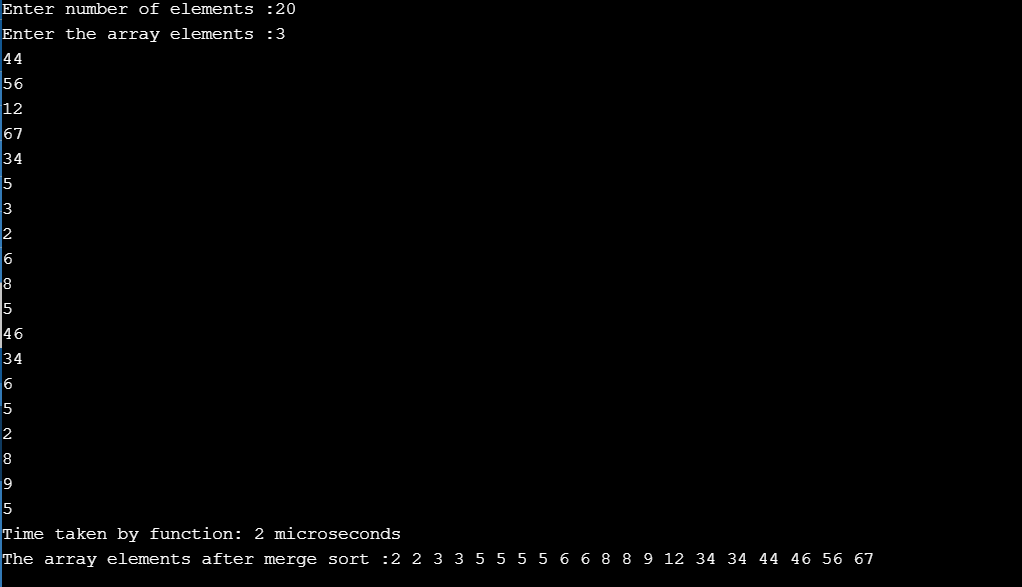
        cout<<A[i]<<" ";

    }

    cout<<endl;

}

Screenshot:



Kompleksitas Algoritma merge sort adalah O(n lg n). Cari tahu kecepatan komputer Anda dalam memproses program. Hitung berapa running time yang dibutuhkan apabila input untuk merge sort-nya adalah 20?

Untuk di program hasilnya : 2 microseconds

Tapi jika sesuai dengan O -> T(26

1. Selection Sort

Source code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Nama Program    : Selection sort

Nama            : Rifaa' Zalfaa' Fakhriyyah

NPM             : 140810170031

\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include<conio.h>

using namespace std;

int data[100],data2[100];

int n;

void tukar(int a, int b)

{

    int t;

    t = data[b];

    data[b] = data[a];

    data[a] = t;

}

void selection\_sort()

{

    int pos,i,j;

    for(i=1;i<=n-1;i++)

    {

     pos = i;

     for(j = i+1;j<=n;j++)

     {

         if(data[j] < data[pos]) pos = j;

        }

if(pos != i) tukar(pos,i);

}

}

int main()

{

    cout << "\n=====================================";

    cout<<"\nMasukkan Jumlah Data : ";cin>>n;

    cout << "\n-------------------------------------" << endl;

    for(int i=1;i<=n;i++)

    {

        cout<<"Masukkan data ke-"<<i<<" : ";

        cin>>data[i];

        data2[i]=data[i];

    }

    selection\_sort();

    cout << "\n-------------------------------------" << endl;

    cout<<"Data Setelah di Sort : ";

    for(int i=1; i<=n; i++)

    {

       cout<<" "<<data[i];

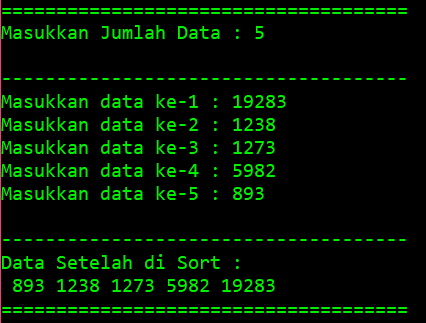
    }

    cout << "\n=====================================\n";

    getch();

}

Screenshot:



Menentukan T(n):

Oleh Karena itu:

Karena ,Maka

1. Insertion Sort

Source code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Nama Program  : Insertion sort

Nama      : Rifaa' Zalfaa' Fakhriyyah

NPM       : 140810170031

\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include <conio.h>

using namespace std;

int data[100],data2[100],n;

void insertion\_sort()

{

  int temp,i,j;

  for(i=1;i<=n;i++){

   temp = data[i];

    j = i -1;

   while(data[j]>temp && j>=0){

      data[j+1] = data[j];

     j--;

   }

   data[j+1] = temp;

  }

}

int main()

{

  cout << "\n=================================="<<endl;

  cout<<"Masukkan Jumlah Data : "; cin>>n;

  cout<<endl;

  cout << "\n----------------------------------" << endl;

  for(int i=1;i<=n;i++)

  {

   cout<<"Masukkan data ke-"<<i<<" : ";

   cin>>data[i];

   data2[i]=data[i];

  }

  cout << "\n----------------------------------" << endl;

  insertion\_sort();

  cout<<"\nData Setelah di Sort : "<<endl;

  for(int i=1; i<=n; i++)

  {

   cout<<data[i]<<" ";

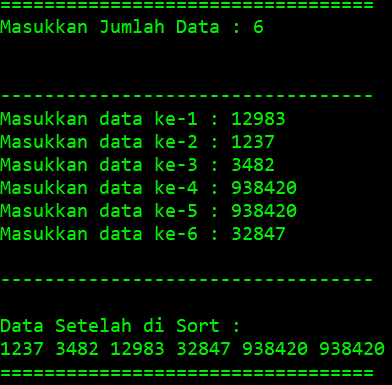
  }

  cout << "\n=================================="<<endl;

  getch();

}

Screenshot:



Menentukan T(n):

T(n) =

1. Bubble sort

Source code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Nama Program    : Insertion sort

Nama            : Rifaa' Zalfaa' Fakhriyyah

NPM             : 140810170031

\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include <conio.h>

using namespace std;

int main(){

    int arr[100],n,temp;

    cout << "\n================================================"<<endl;

    cout<<"Massukan banyak elemen yang akan diinputkan : ";cin>>n;

    cout << "\n------------------------------------------------" << endl;

    for(int i=0;i<n;++i){

        cout<<"Masukkan Elemen ke-"<<i+1<<" : ";cin>>arr[i];

    }

    for(int i=1;i<n;i++){

        for(int j=0;j<(n-1);j++){

            if(arr[j]>arr[j+1]){

                temp=arr[j];

                arr[j]=arr[j+1];

                arr[j+1]=temp;

            }

        }

    }

    cout << "------------------------------------------------" << endl;

    cout<<"\nHasil dari Bubble Sort : "<<endl;

    for(int i=0;i<n;i++){

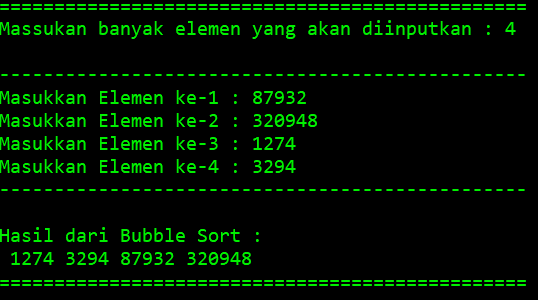
        cout<<" "<<arr[i];

    }

    cout << "\n================================================"<<endl;

}

Screenshot:



Menentukan T(n):