Design dan Analisis Algoritma

Nama Lengkap : **RIFANDY ARNAS**

NIM/NPM : **232310001**

Kelas : **TI – 23 – PA (Lab 2)**

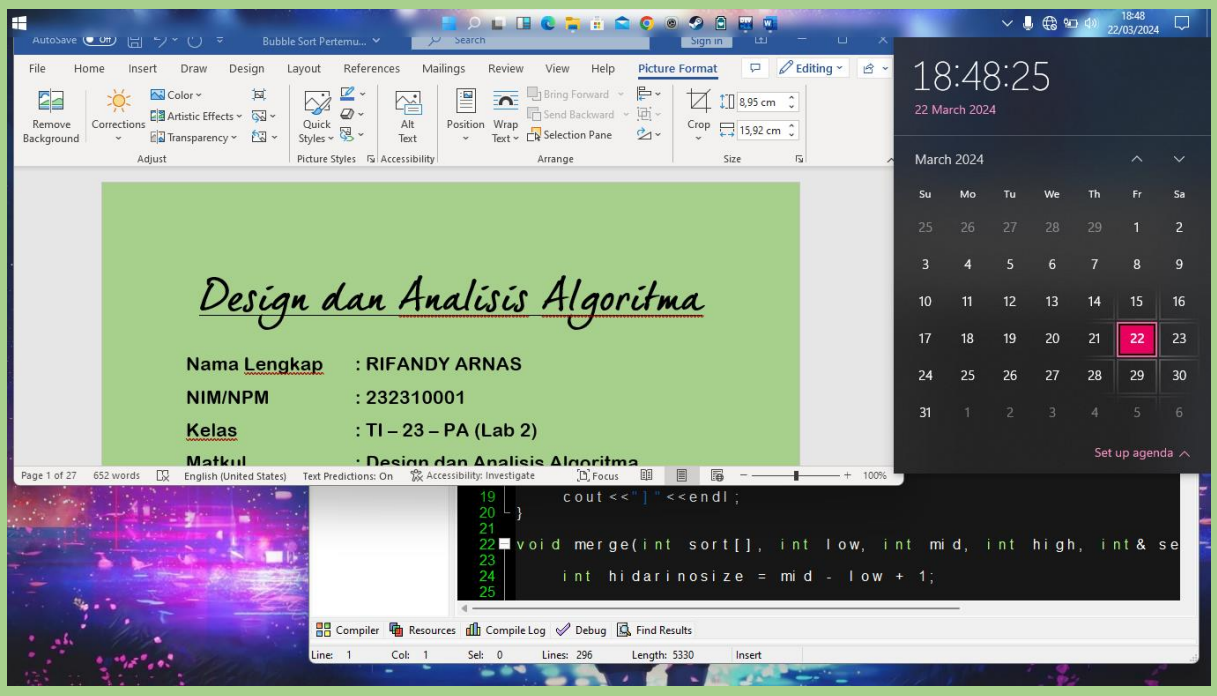
Matkul : **Design dan Analisis Algoritma**

PERTEMUAN II

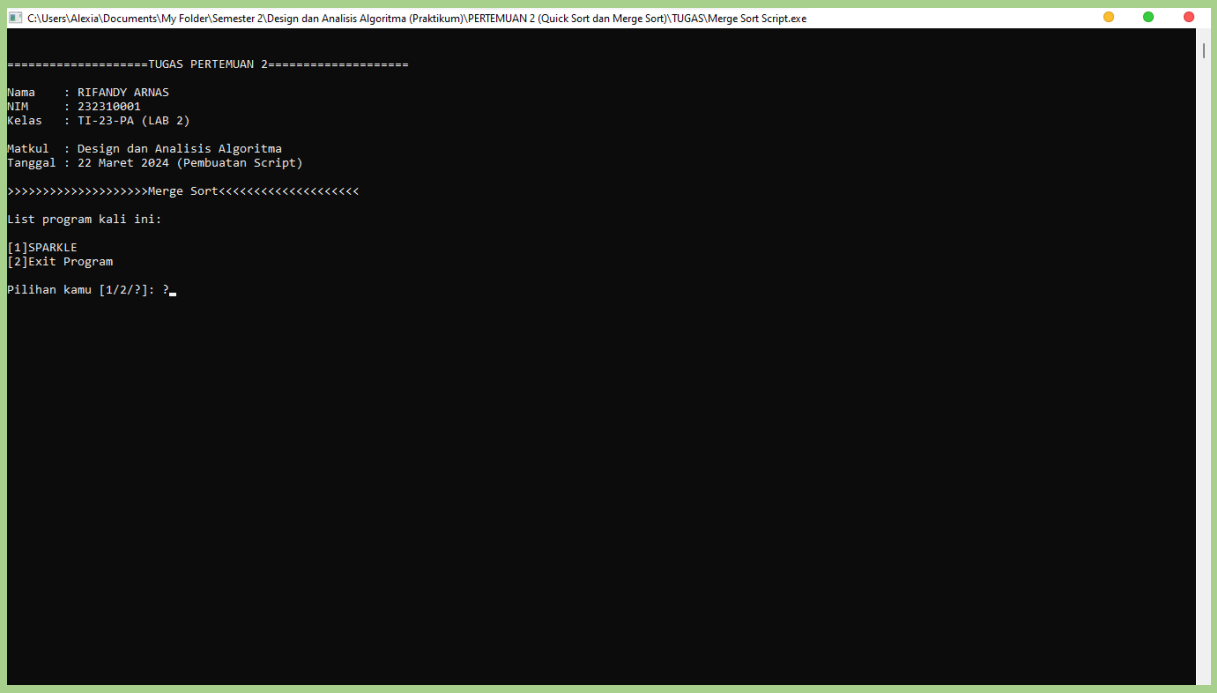
[TUGAS]

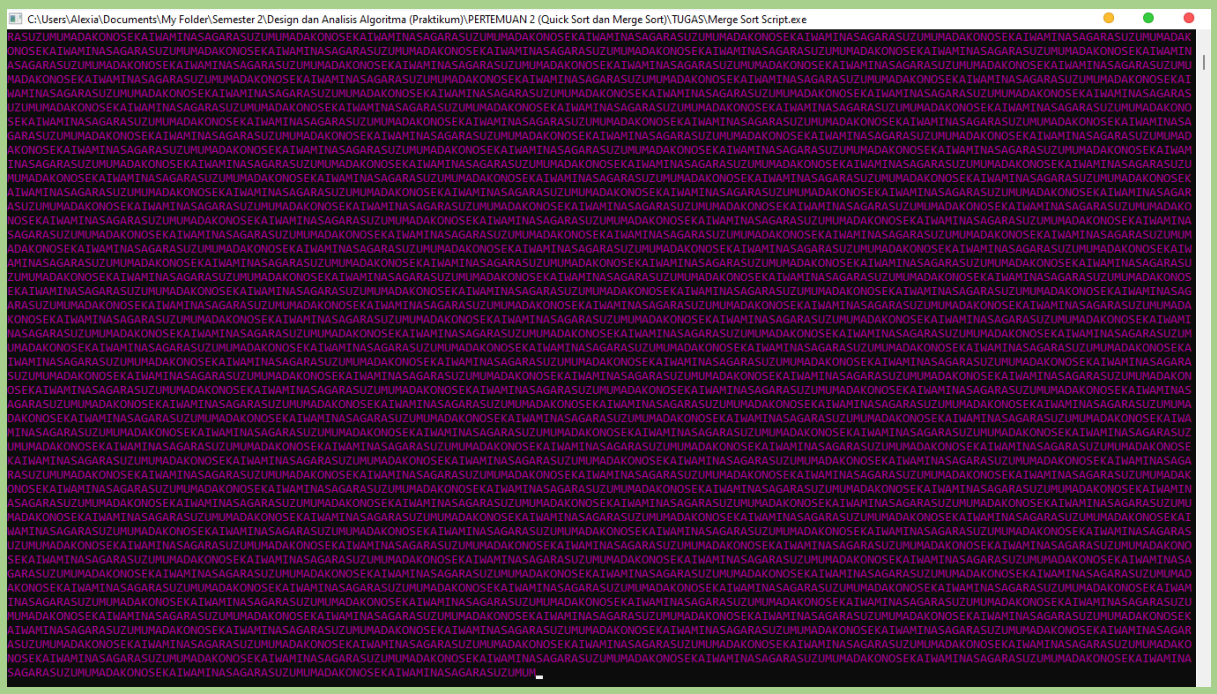
>>>>>>>>>>Quick Sort dan Merge Sort<<<<<<<<<<

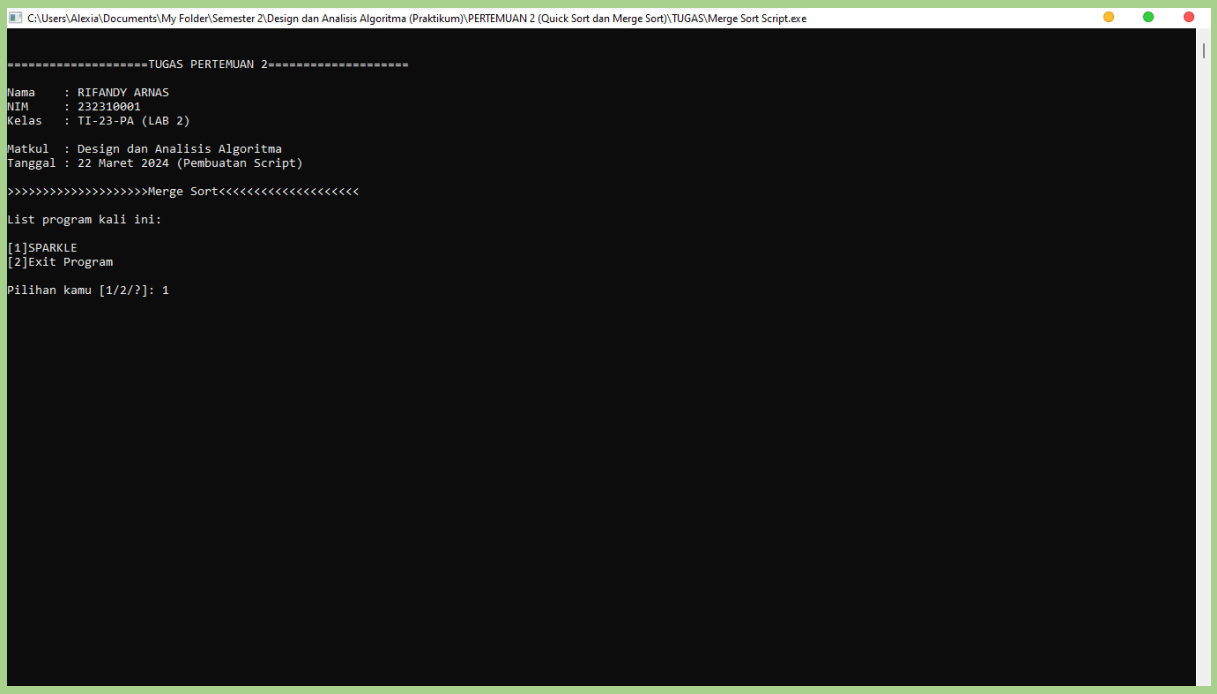
Berikut adalah screnshoot waktu dan tanggal pembuatan script

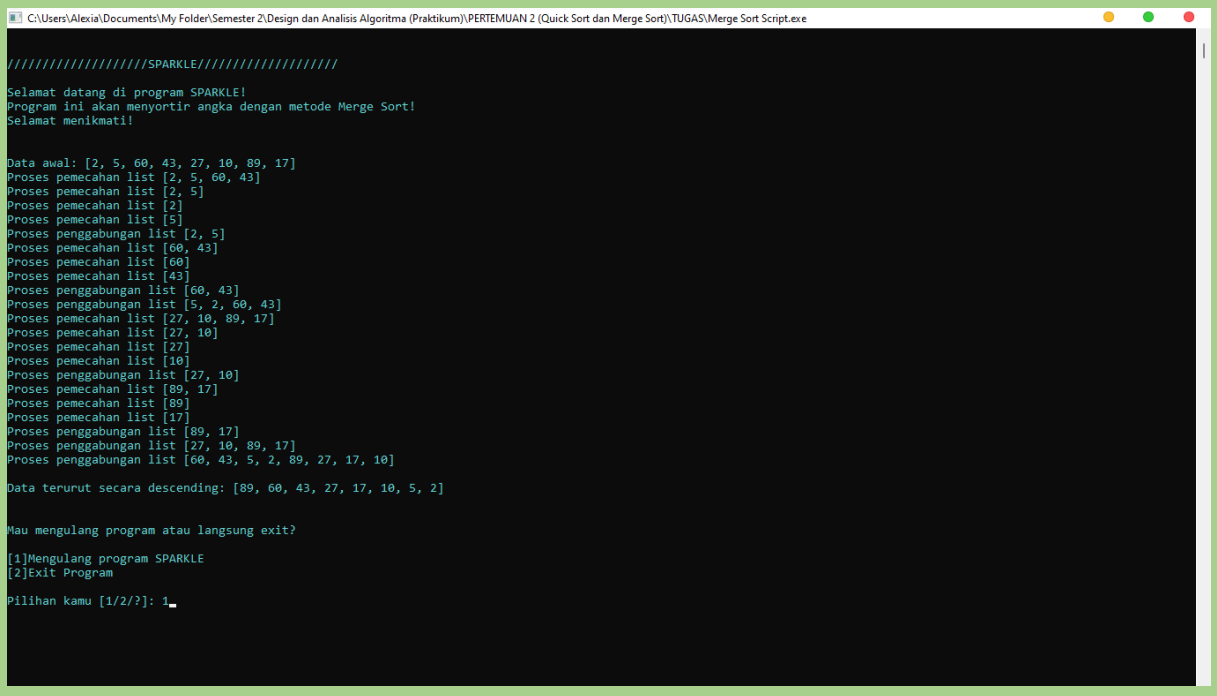


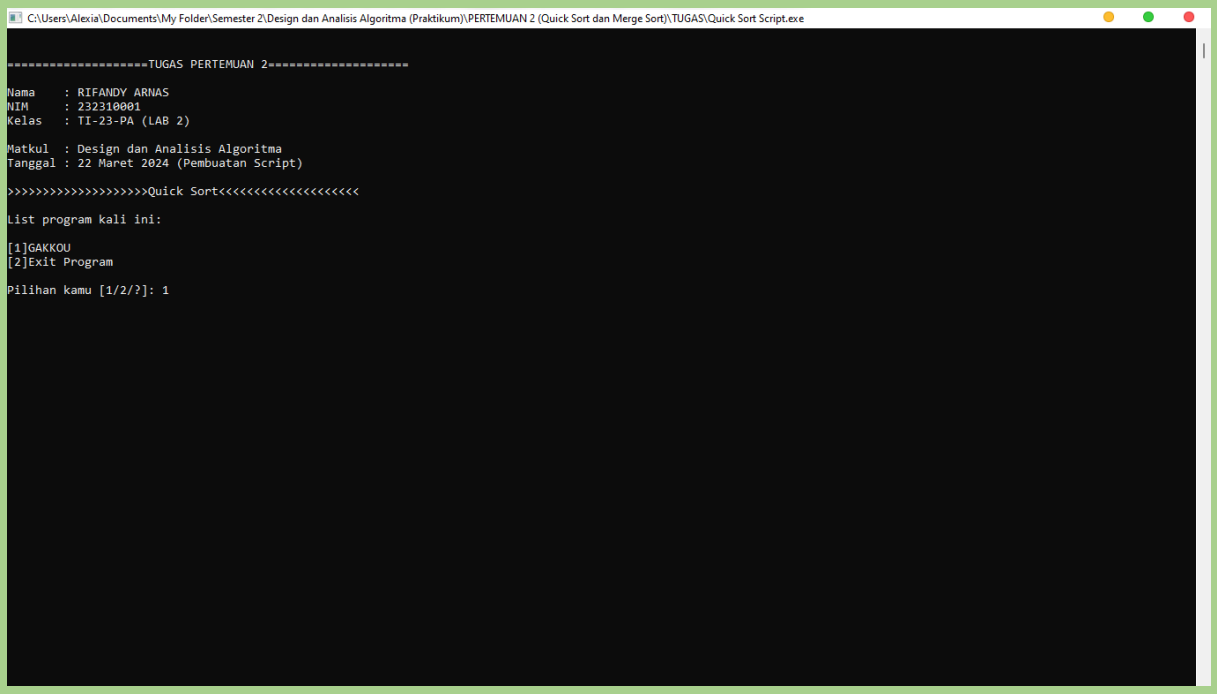
Berikut adalah screnshoot Output dari pembuatan program kali ini

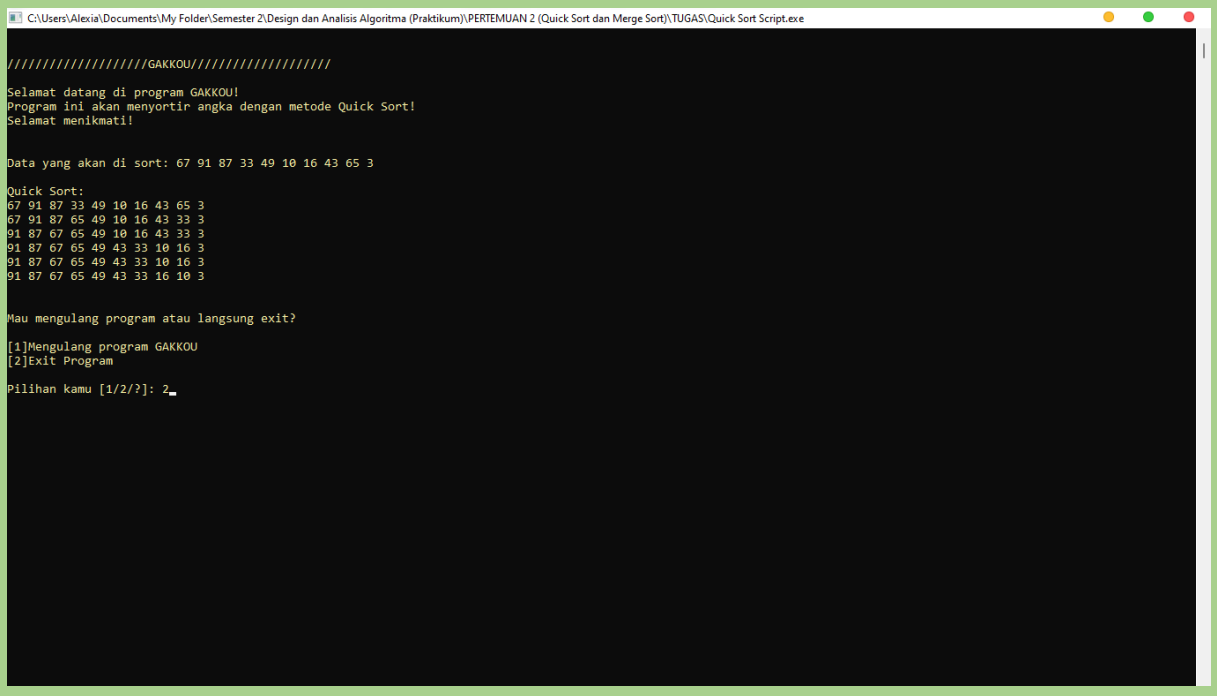


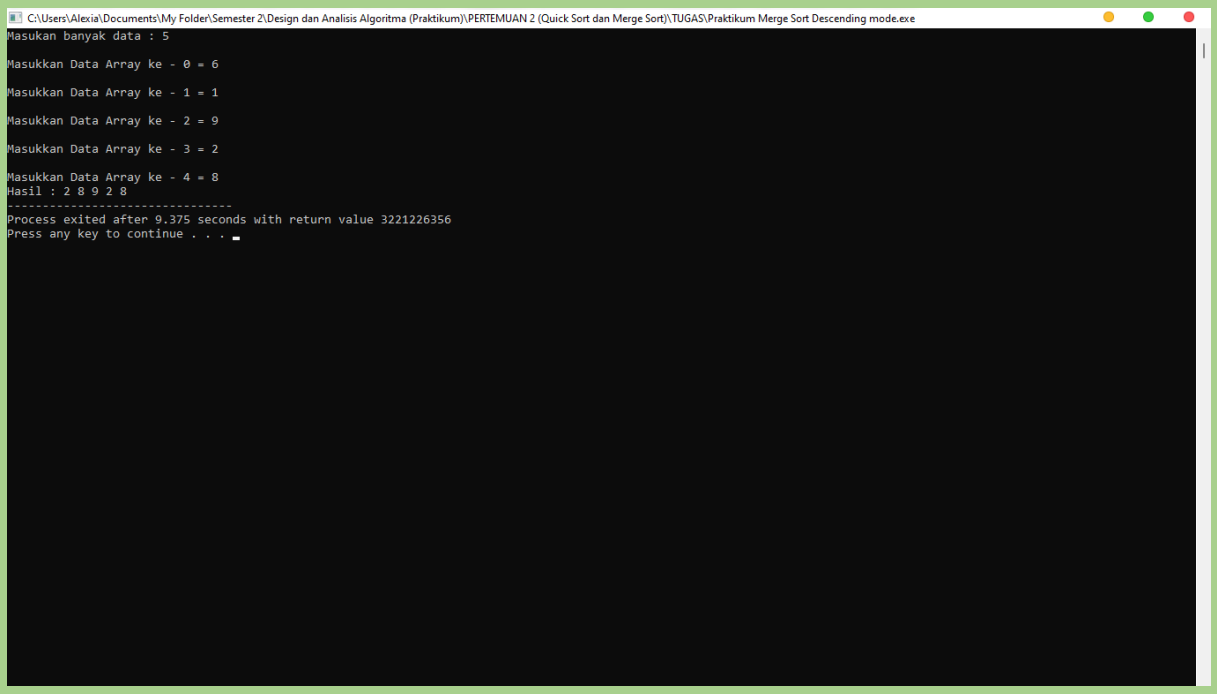



C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Merge Sort Script.exe
==========TUGAS PERTEMUAN 2==========
Nama : RIFANDY ARNAS
NIM : 232310001
Kelas : TI-23-PA (LAB 2)
Matkul : Design dan Analisis Algoritma
Tanggal : 22 Maret 2024 (Pembuatan Script)
>>>>>>>>>>>>>>>>>>>Merge Sort<<<<<<<<<<<<<<<<<<<<
List program kali ini:
[1]SPARKLE
[2]Exit Program
Pilihan kamu [1/2/?]: 1

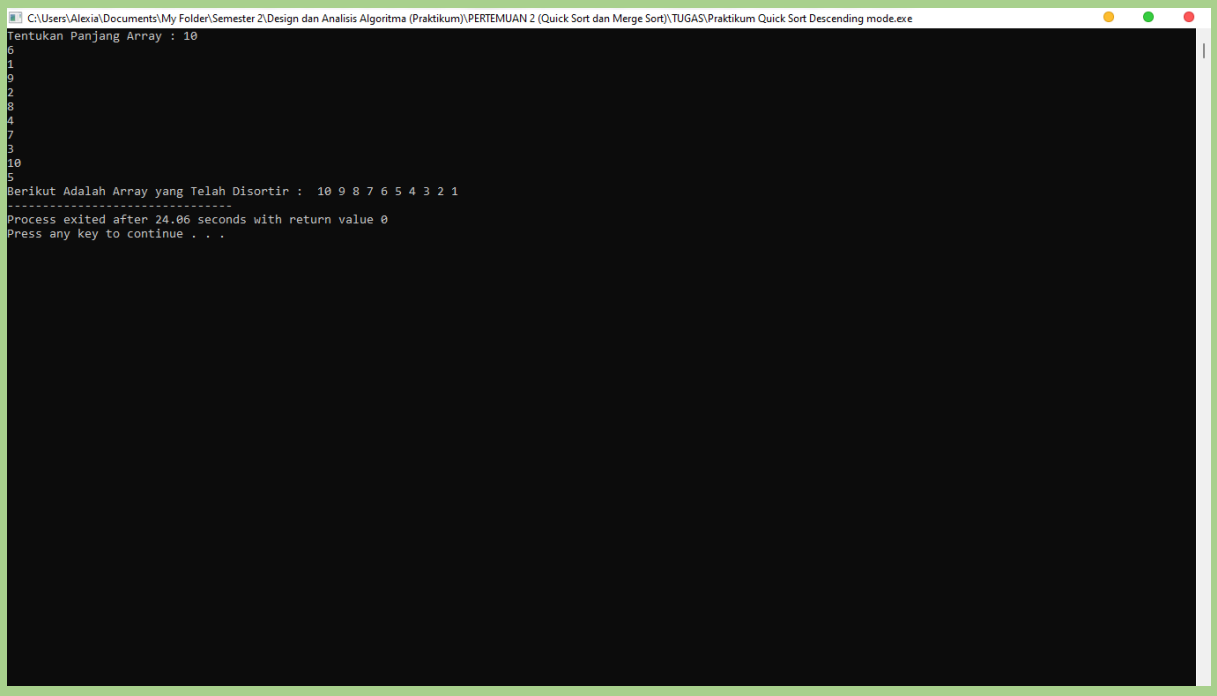

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Merge Sort Script.exe
//////////////////SPARKLE//////////////////
Selamat datang di program SPARKLE!
Program ini akan menyortir angka dengan metode Merge Sort!
Selamat menikmati!
Data awal: [2, 5, 60, 43, 27, 10, 89, 17]
Proses pemecahan list [2, 5, 60, 43]
Proses pemecahan list [2, 5]
Proses pemecahan list [2]
Proses pemecahan list [5]
Proses penggabungan list [2, 5]
Proses pemecahan list [60, 43]
Proses pemecahan list [60]
Proses pemecahan list [43]
Proses penggabungan list [60, 43]
Proses penggabungan list [5, 2, 60, 43]
Proses pemecahan list [27, 10, 89, 17]
Proses pemecahan list [27, 10]
Proses pemecahan list [27]
Proses pemecahan list [10]
Proses penggabungan list [27, 10]
Proses pemecahan list [89, 17]
Proses pemecahan list [89]
Proses pemecahan list [17]
Proses penggabungan list [89, 17]
Proses penggabungan list [27, 10, 89, 17]
Proses penggabungan list [60, 43, 5, 2, 89, 27, 17, 10]
Data terurut secara descending: [89, 60, 43, 27, 17, 10, 5, 2]
Mau mengulang program atau langsung exit?
[1]Mengulang program SPARKLE
[2]Exit Program
Pilihan kamu [1/2/?]: 1


C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Quick Sort Script.exe
==========TUGAS PERTEMUAN 2==========
Nama : RIFANDY ARNAS
NIM : 232310001
Kelas : TI-23-PA (LAB 2)
Matkul : Design dan Analisis Algoritma
Tanggal : 22 Maret 2024 (Pembuatan Script)
>>>>>>>>>>>>>>>>>>>Quick Sort<<<<<<<<<<<<<<<<<<<<
List program kali ini:
[1]GAKKOU
[2]Exit Program
Pilihan kamu [1/2/?]: 1

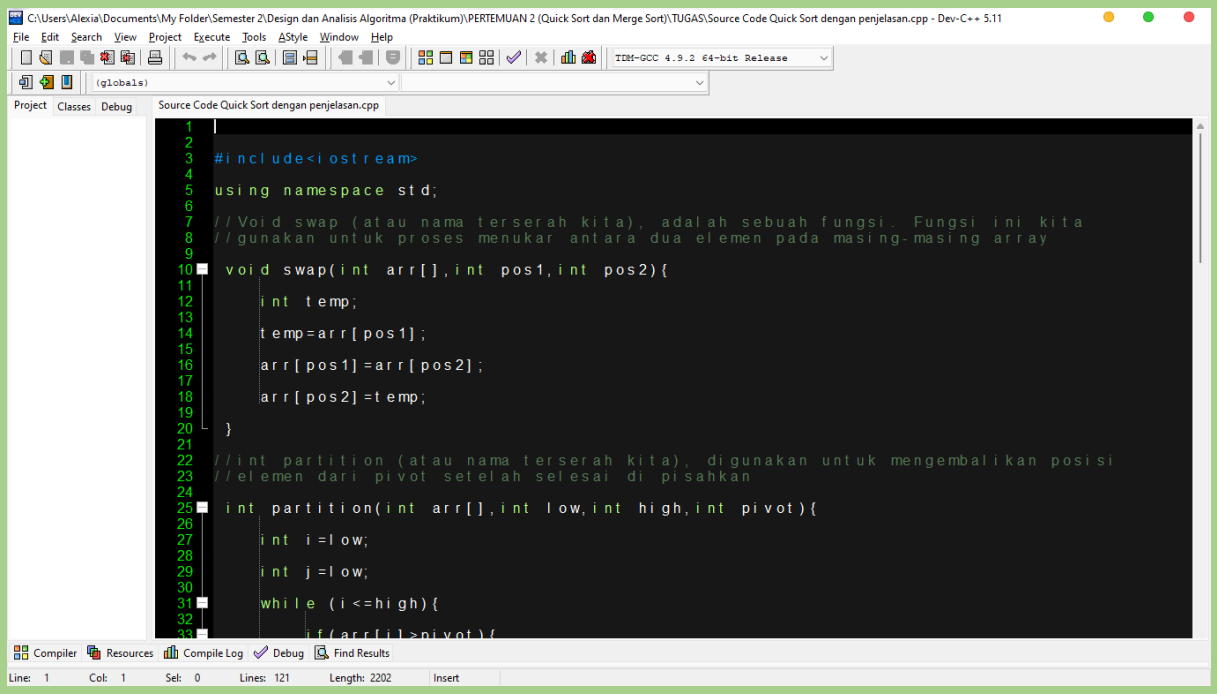

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Quick Sort Script.exe
//////////////////GAKKOU//////////////////
Selamat datang di program GAKKOU!
Program ini akan menyortir angka dengan metode Quick Sort!
Selamat menikmati!
Data yang akan di sort: 67 91 87 33 49 10 16 43 65 3
Quick Sort:
67 91 87 33 49 10 16 43 65 3
67 91 87 65 49 10 16 43 33 3
91 87 67 65 49 10 16 43 33 3
91 87 67 65 49 43 33 10 16 3
91 87 67 65 49 43 33 10 16 3
91 87 67 65 49 43 33 16 10 3
Mau mengulang program atau langsung exit?
[1]Mengulang program GAKKOU
[2]Exit Program
Pilihan kamu [1/2/?]: 2

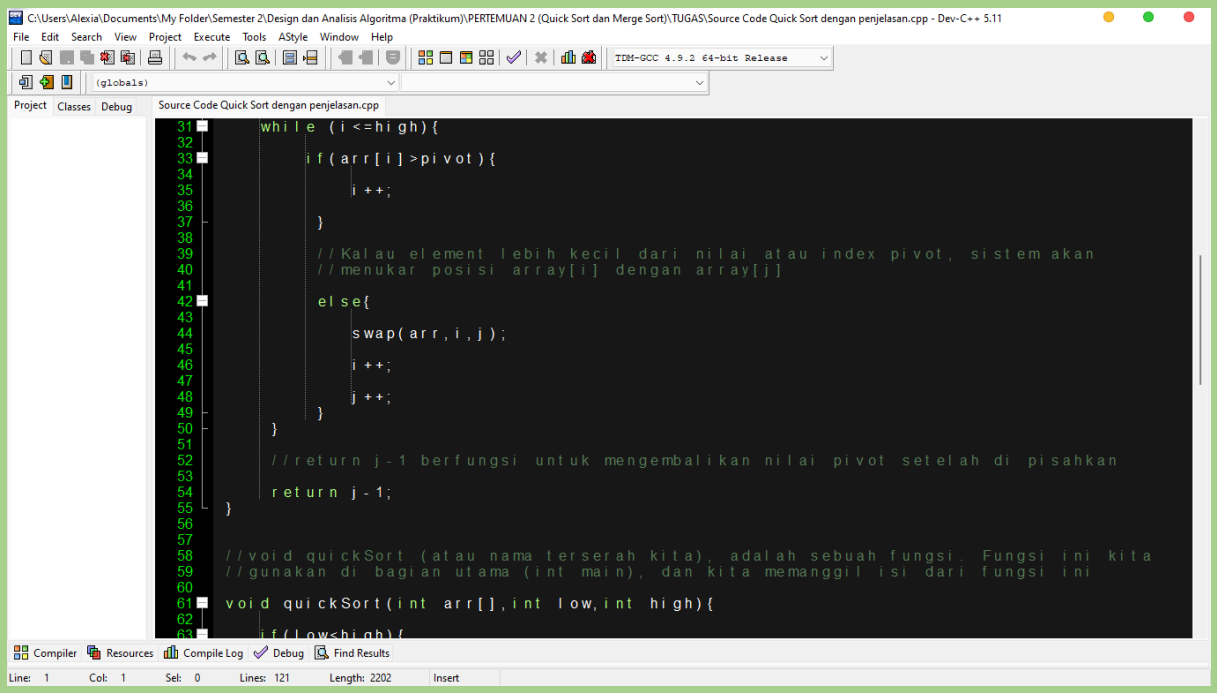

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Praktikum Merge Sort Descending mode.exe
Masukan banyak data : 5
Masukkan Data Array ke - 0 = 6
Masukkan Data Array ke - 1 = 1
Masukkan Data Array ke - 2 = 9
Masukkan Data Array ke - 3 = 2
Masukkan Data Array ke - 4 = 8
Hasil : 2 8 9 2 8

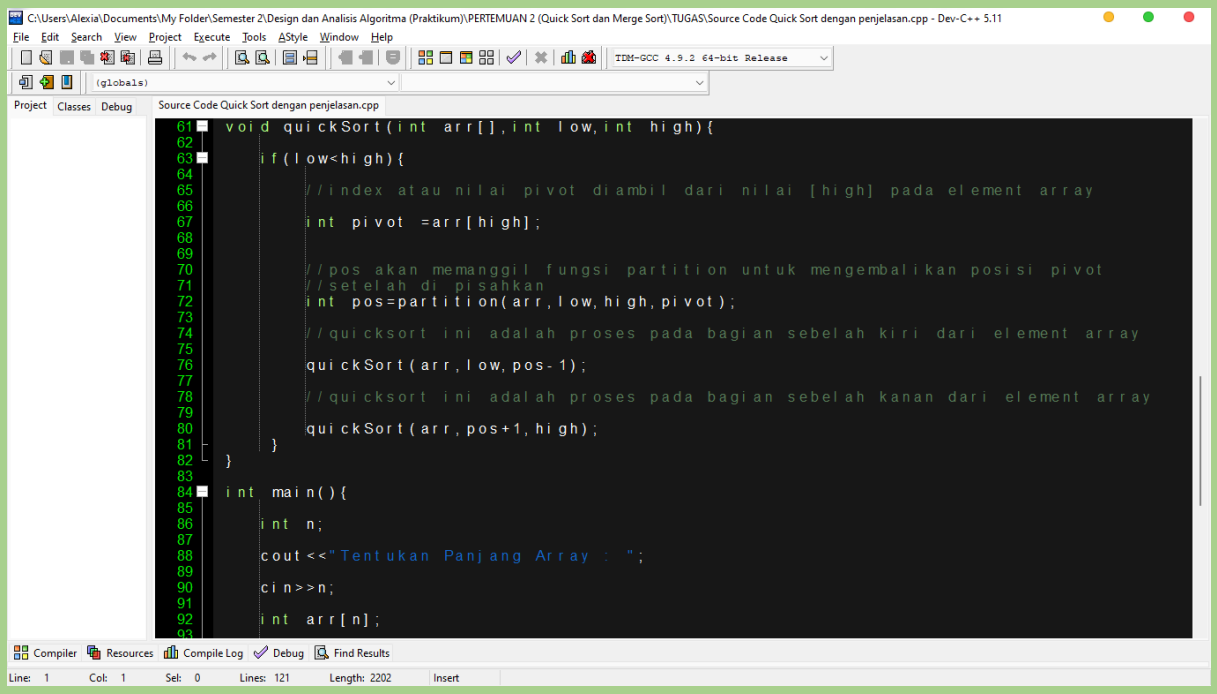
Process exited after 9.375 seconds with return value 3221226356
Press any key to continue . . .

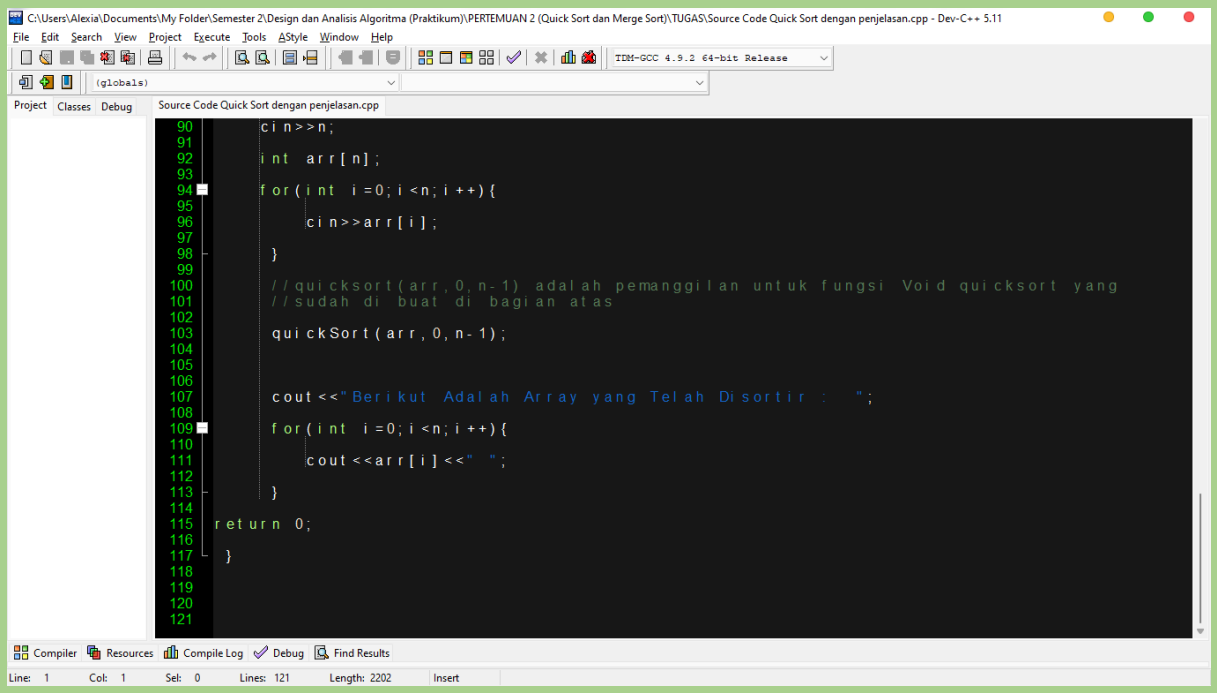

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Praktikum Quick Sort Descending mode.exe
Tentukan Panjang Array : 10
6
1
9
2
8
4
7
3
10
5
Berikut Adalah Array yang Telah Disortir : 10 9 8 7 6 5 4 3 2 1

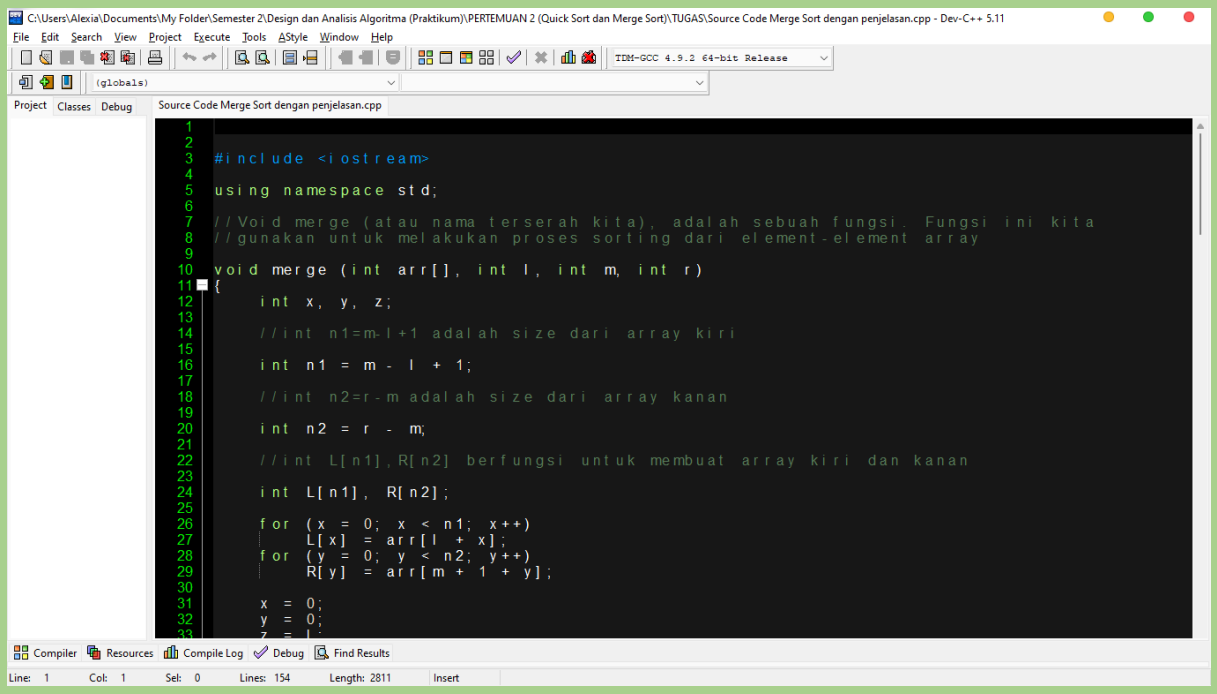
Process exited after 24.06 seconds with return value 0
Press any key to continue . . .

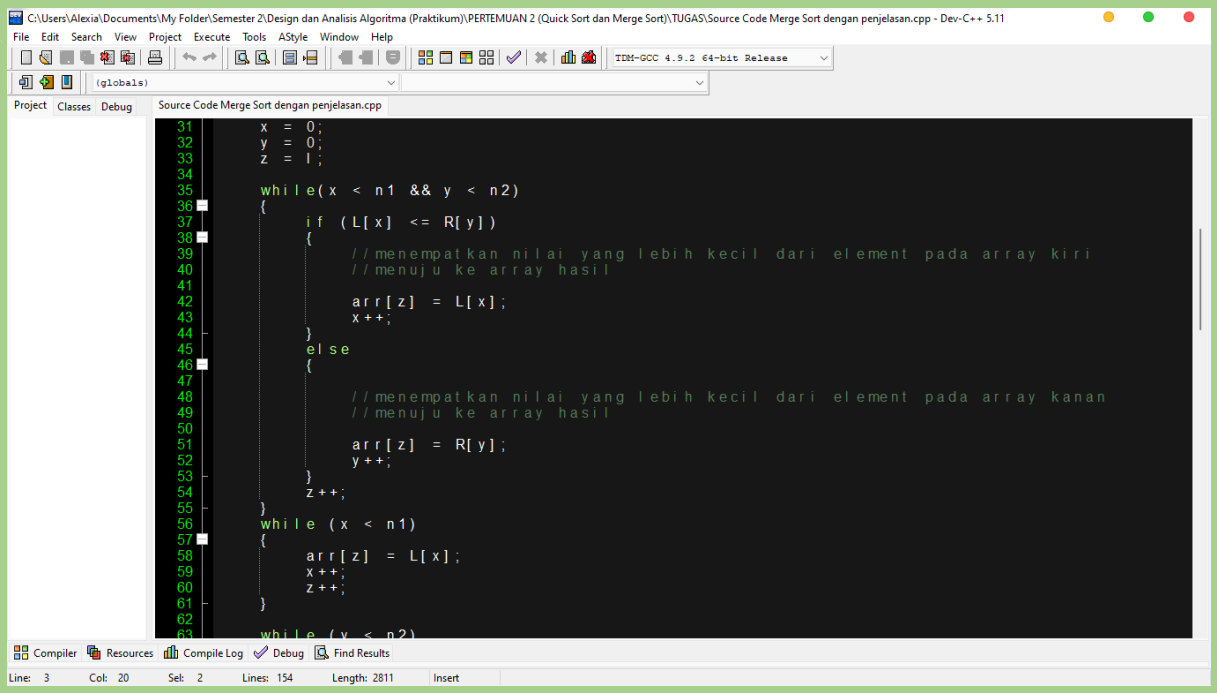

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Quick Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Quick Sort dengan penjelasan.cpp
#include<iostream>
using namespace std;
//Void swap (atau nama terserah kita), adalah sebuah fungsi. Fungsi ini kita
//gunakan untuk proses menukar antara dua elemen pada masing-masing array
void swap(int arr[],int pos1,int pos2){
int temp;
temp=arr[pos1];
arr[pos1]=arr[pos2];
arr[pos2]=temp;
}
//int partition (atau nama terserah kita), digunakan untuk mengembalikan posisi
//elemen dari pivot setelah selesai di pisahkan
int partition(int arr[],int low,int high,int pivot){
int i=low;
int j=low;
while (i<=high){
if(arr[i]>pivot){
Compiler Resources Compile Log Debug Find Results
Line: 1 Col: 1 Sel: 0 Lines: 121 Length: 2202 Insert

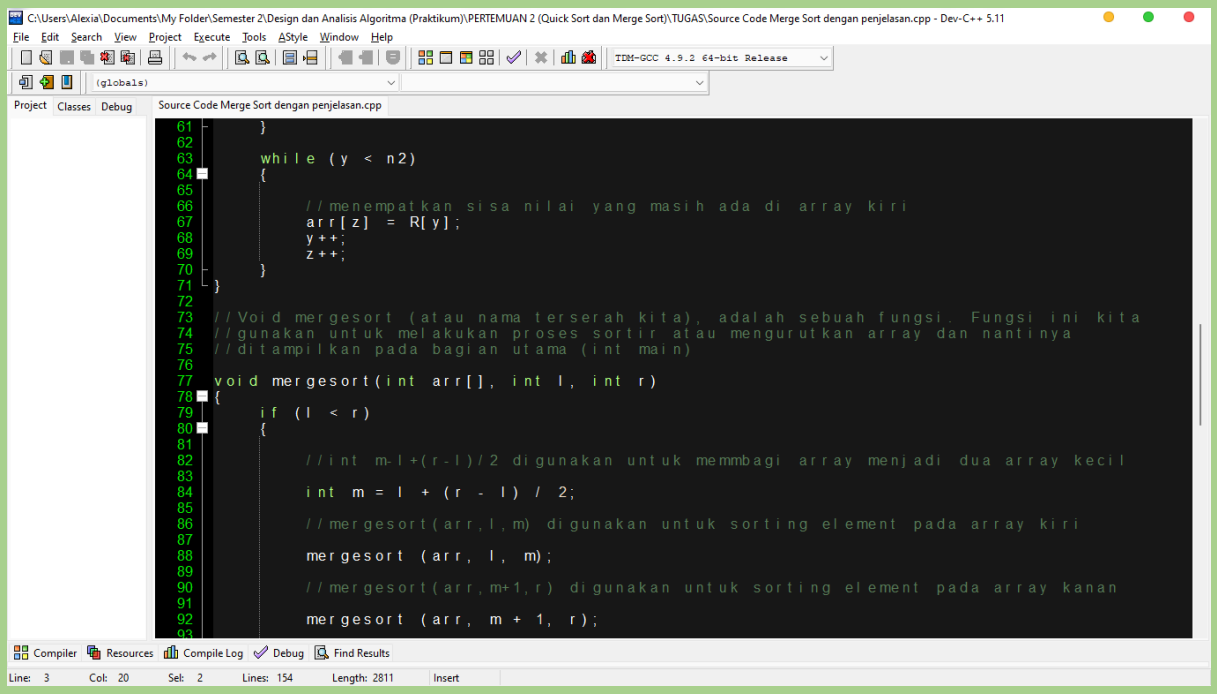

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Quick Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Quick Sort dengan penjelasan.cpp
while (i<=high){
if(arr[i]>pivot){
i++;
}
//Kalau element lebih kecil dari nilai atau index pivot, sistem akan
//menukar posisi array[i] dengan array[j]
else{
swap(arr,i,j);
i++;
j++;
}
}
//return j-1 berfungsi untuk mengembalikan nilai pivot setelah di pisahkan
return j-1;
}
//void quickSort (atau nama terserah kita), adalah sebuah fungsi. Fungsi ini kita
//gunakan di bagian utama (int main), dan kita memanggil isi dari fungsi ini
void quickSort(int arr[],int low,int high){
if(low<high){
Compiler Resources Compile Log Debug Find Results
Line: 1 Col: 1 Sel: 0 Lines: 121 Length: 2202 Insert

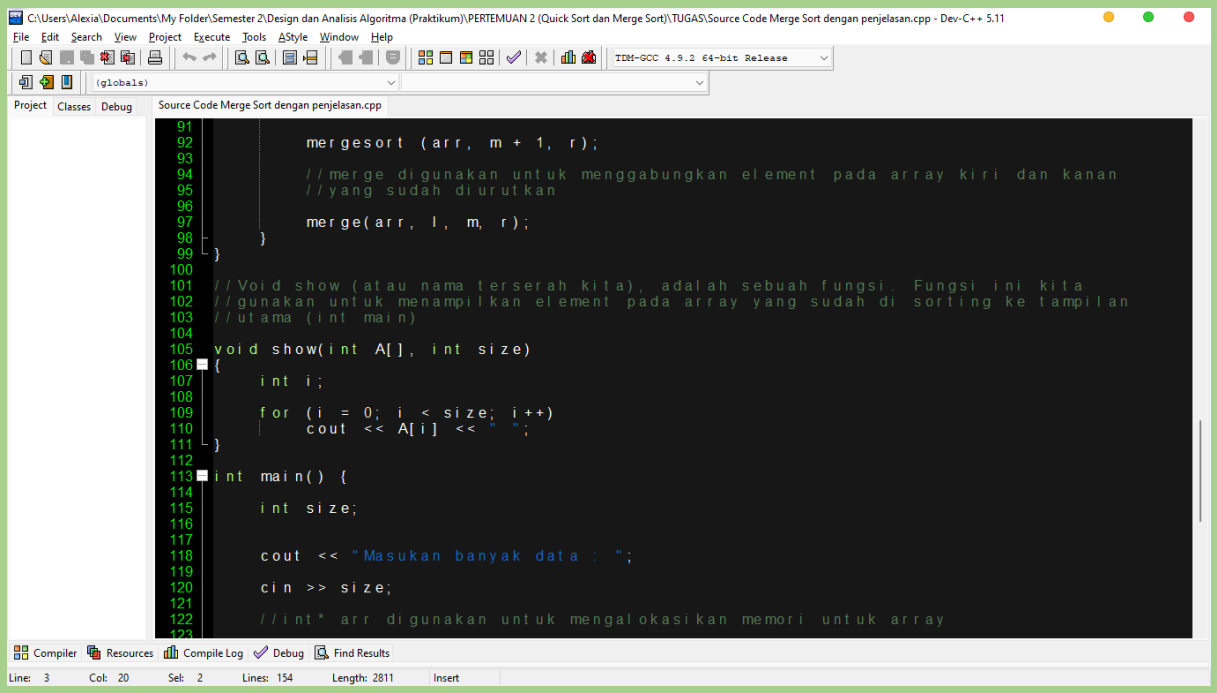

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Quick Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Quick Sort dengan penjelasan.cpp
void quickSort(int arr[],int low,int high){
if(low<high){
//index atau nilai pivot diambil dari nilai [high] pada element array
int pivot =arr[high];
//pos akan memanggil fungsi partition untuk mengembalikan posisi pivot
//setelah di pisahkan
int pos=partition(arr,low,high,pivot);
//quicksort ini adalah proses pada bagian sebelah kiri dari element array
quickSort(arr,low,pos-1);
//quicksort ini adalah proses pada bagian sebelah kanan dari element array
quickSort(arr,pos+1,high);
}
}
int main(){
int n;
cout<<"Tentukan Panjang Array : ";
cin>>n;
int arr[n];
Compiler Resources Compile Log Debug Find Results
Line: 1 Col: 1 Sel: 0 Lines: 121 Length: 2202 Insert

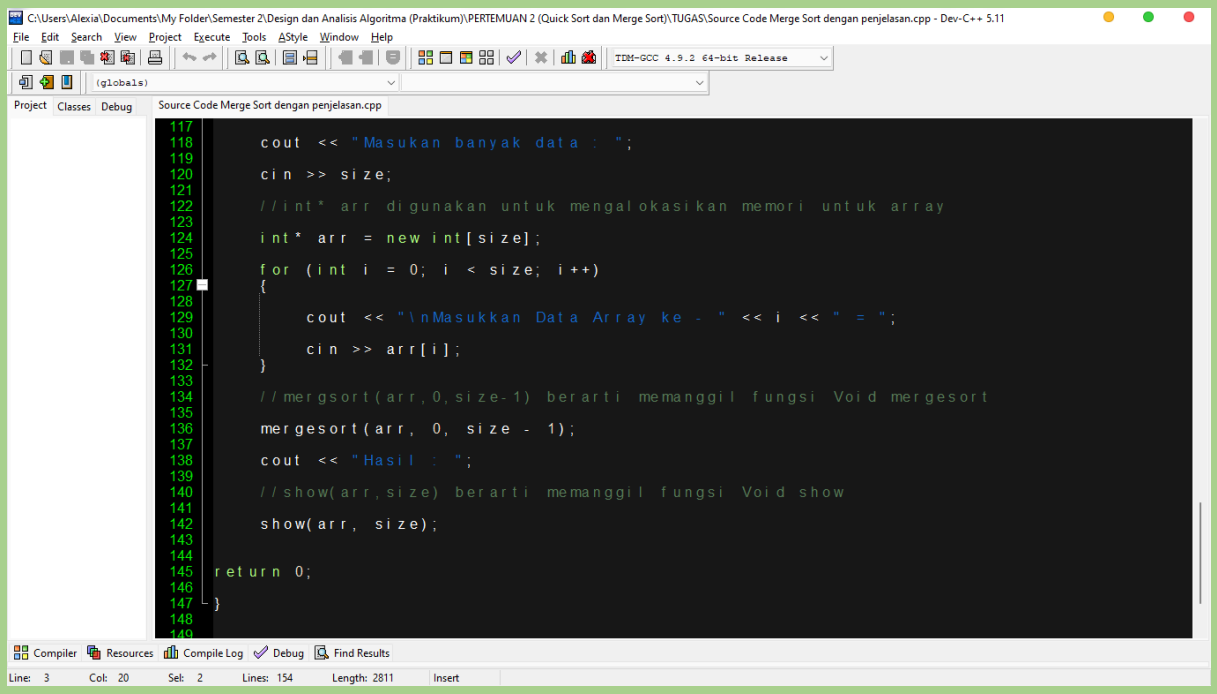

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Quick Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Quick Sort dengan penjelasan.cpp
cin>>n;
int arr[n];
for(int i=0;i<n;i++){
cin>>arr[i];
}
//quicksort(arr,0,n-1) adalah pemanggilan untuk fungsi Void quicksort yang
//sudah di buat di bagian atas
quickSort(arr,0,n-1);
cout<<"Berikut Adalah Array yang Telah Disortir : ";
for(int i=0;i<n;i++){
cout<<arr[i]<<" ";
}
return 0;
}
Compiler Resources Compile Log Debug Find Results
Line: 1 Col: 1 Sel: 0 Lines: 121 Length: 2202 Insert

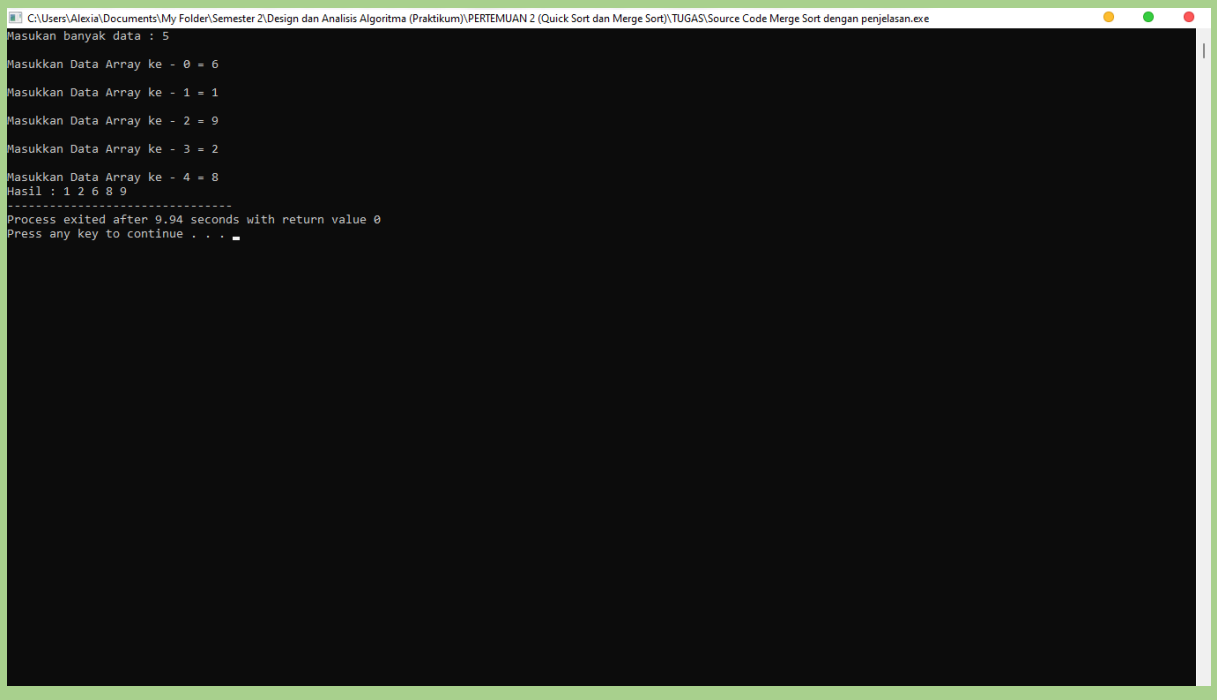

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Merge Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Merge Sort dengan penjelasan.cpp
#include <iostream>
using namespace std;
//Void merge (atau nama terserah kita), adalah sebuah fungsi. Fungsi ini kita
//gunakan untuk melakukan proses sorting dari element-element array
void merge (int arr[], int l, int m, int r)
{
int x, y, z;
//int n1=m-l+1 adalah size dari array kiri
int n1 = m - l + 1;
//int n2=r-m adalah size dari array kanan
int n2 = r - m;
//int L[n1],R[n2] berfungsi untuk membuat array kiri dan kanan
int L[n1], R[n2];
for (x = 0; x < n1; x++)
L[x] = arr[l + x];
for (y = 0; y < n2; y++)
R[y] = arr[m + 1 + y];
x = 0;
y = 0;
Compiler Resources Compile Log Debug Find Results
Line: 1 Col: 1 Sel: 0 Lines: 154 Length: 2811 Insert


C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Merge Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Merge Sort dengan penjelasan.cpp
x = 0;
y = 0;
z = l;
while(x < n1 && y < n2)
{
if (L[x] <= R[y])
{
//menempatkan nilai yang lebih kecil dari element pada array kiri
//menuju ke array hasil
arr[z] = L[x];
x++;
}
else
{
//menempatkan nilai yang lebih kecil dari element pada array kanan
//menuju ke array hasil
arr[z] = R[y];
y++;
}
z++;
}
while (x < n1)
{
arr[z] = L[x];
x++;
z++;
}
while (y < n2)
Compiler Resources Compile Log Debug Find Results
Line: 3 Col: 20 Sel: 2 Lines: 154 Length: 2811 Insert

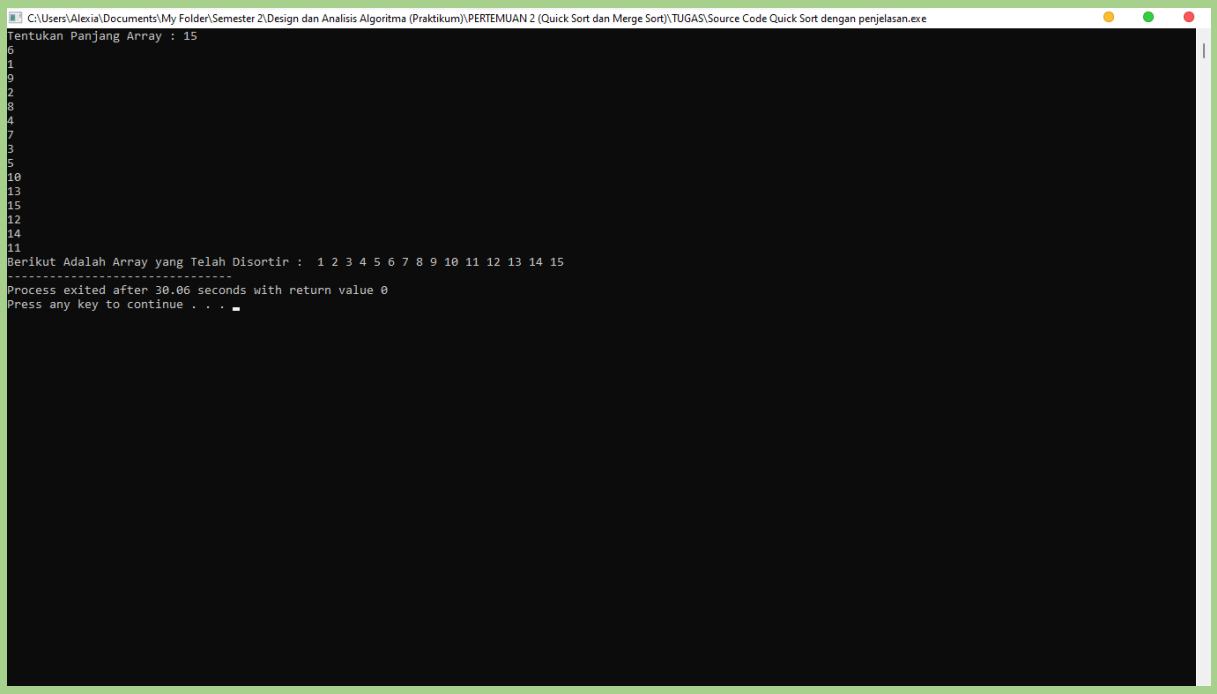

C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Merge Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Merge Sort dengan penjelasan.cpp
61 }
62
63 while (y < n2)
64 {
65
66 //menempatkan sisa nilai yang masih ada di array kiri
67 arr[z] = R[y];
68 y++;
69 z++;
70 }
71 }
72
73 //Void mergesort (atau nama terserah kita), adalah sebuah fungsi. Fungsi ini kita
74 //gunakan untuk melakukan proses sortir atau mengurutkan array dan nantinya
75 //ditampilkan pada bagian utama (int main)
76
77 void mergesort(int arr[], int l, int r)
78 {
79 if (l < r)
80 {
81
82 //int m-l+(r-l)/2 digunakan untuk memmbagi array menjadi dua array kecil
83
84 int m = l + (r - l) / 2;
85
86 //mergesort(arr,l,m) digunakan untuk sorting element pada array kiri
87
88 mergesort (arr, l, m);
89
90 //mergesort(arr,m+1,r) digunakan untuk sorting element pada array kanan
91
92 mergesort (arr, m + 1, r);
93
Compiler Resources Compile Log Debug Find Results
Line: 3 Col: 20 Sel: 2 Lines: 154 Length: 2811 Insert


C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Merge Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Merge Sort dengan penjelasan.cpp
91
92 mergesort (arr, m + 1, r);
93
94 //merge digunakan untuk menggabungkan element pada array kiri dan kanan
95 //yang sudah diurutkan
96
97 merge(arr, l, m, r);
98 }
99 }
100
101 //Void show (atau nama terserah kita), adalah sebuah fungsi. Fungsi ini kita
102 //gunakan untuk menampilkan element pada array yang sudah di sorting ke tampilan
103 //utama (int main)
104
105 void show(int A[], int size)
106 {
107 int i;
108
109 for (i = 0; i < size; i++)
110 cout << A[i] << " ";
111 }
112
113 int main() {
114
115 int size;
116
117
118 cout << "Masukan banyak data : ";
119
120 cin >> size;
121
122 //int* arr digunakan untuk mengalokasikan memori untuk array
123
Compiler Resources Compile Log Debug Find Results
Line: 3 Col: 20 Sel: 2 Lines: 154 Length: 2811 Insert


C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Merge Sort dengan penjelasan.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Source Code Merge Sort dengan penjelasan.cpp
117 cout << "Masukan banyak data : ";
118
119 cin >> size;
120
121 //int* arr digunakan untuk mengalokasikan memori untuk array
122
123 int* arr = new int[size];
124
125 for (int i = 0; i < size; i++)
126 {
127
128 cout << "\nMasukkan Data Array ke - " << i << " = ";
129
130 cin >> arr[i];
131 }
132
133 //mergsort(arr,0,size-1) berarti memanggil fungsi Void mergesort
134
135 mergesort(arr, 0, size - 1);
136
137 cout << "Hasil : ";
138
139 //show(arr,size) berarti memanggil fungsi Void show
140
141 show(arr, size);
142
143
144 return 0;
145 }
146
147
148
149
Compiler Resources Compile Log Debug Find Results
Line: 3 Col: 20 Sel: 2 Lines: 154 Length: 2811 Insert


C:\Users\Alexia\Documents\My Folder\Semester 2\Design dan Analisis Algoritma (Praktikum)\PERTEMUAN 2 (Quick Sort dan Merge Sort)\TUGAS\Source Code Merge Sort dengan penjelasan.exe
Masukan banyak data : 5
Masukkan Data Array ke - 0 = 6
Masukkan Data Array ke - 1 = 1
Masukkan Data Array ke - 2 = 9
Masukkan Data Array ke - 3 = 2
Masukkan Data Array ke - 4 = 8
Hasil : 1 2 6 8 9

Process exited after 9.94 seconds with return value 0
Press any key to continue . . .



Berikut adalah kode script yang saya buat

///////Quick Sort kode Script///////

```
#include <iostream>
#include <string>

using namespace std;

void Sorting(int sort[], int size) {

    for (int i = 0; i < 10; i++) {

        cout << sort[i] << " ";

    }
```

```
    cout << endl;
}

int partisi(int sort[], int low, int high) {

    int pivot = sort[high];

    int i = low - 1;

    for (int j = low; j < high; j++) {

        if (sort[j] > pivot) {

            i++;

            swap(sort[i], sort[j]);
        }
    }

    swap(sort[i + 1], sort[high]);

    return i + 1;
}

void QuickSort(int sort[], int low, int high) {

    if (low < high) {
```

```
        int fubuki = partisi(sort, low, high);

        Sorting(sort, high + 1);

        QuickSort(sort, low, fubuki - 1);

        QuickSort(sort, fubuki + 1, high);
    }
}


int main(){

system("COLOR F");

        printf("\n\n====================TUGAS PERTEMUAN
2====================\n\n");

                string nma("RIFANDY ARNAS");
                string nim("232310001");
                string kls("TI-23-PA ");

                kls.append("(LAB 2)");

                string mtkl("Design dan Analisis Algoritma");
                string tgl("22 Maret 2024 ");
```

```
        tgl.append("(Pembuatan Script)");

                cout<<"Nama\t: "<<nma;
                cout<<"\nNIM\t: "<<nim;
                cout<<"\nKelas\t: "<<kls;
                cout<<"\n\nMatkul\t: "<<mtkl;
                cout<<"\nTanggal\t: "<<tgl;

    printf("\n\n>>>>>>>>>>>>>>>>>>>>Quick
Sort<<<<<<<<<<<<<<<<<<<\n\n");

        int pilihan; int kimi_no_na_wa; int tenki_no_ko;

        kimi_no_na_wa=1;

        tenki_no_ko=2;

                cout<<"List program kali ini: ";
                cout<<"\n\n[1]GAKKOU";
                cout<<"\n[2]Exit Program";

                cout<<endl<<endl;

                cout<<"Pilihan kamu [1/2/?]: ";

                cin>>pilihan;

                        if(pilihan==1){
```

```
                goto gakkou;

            }

            else if(pilihan==2){

                system("cls");


cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

                exit(0);

            }

            else{

                system("cls");

                system("COLOR D");


cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;
```

```
                                        while(kimi_no_na_wa <
tenki_no_ko){

        cout<<"MADAKONOSEKAIWAMINASAGARASUZUMU";

                                                kimi_no_na_wa--;

                                        }

                                }

gakkou:

        system("cls");

        system("COLOR E");

        printf("\n\n//////////////////GAKKOU//////////////////\n\n");

                        cout<<"Selamat datang di program GAKKOU!\n";
                        cout<<"Program ini akan menyortir angka dengan
metode Quick Sort!\n";
                        cout<<"Selamat menikmati!";

                        cout<<endl<<endl<<endl;
```

```
            int fortune[] = {67, 91, 87, 33, 49, 10, 16, 43, 65, 3};

        int size = sizeof(fortune) / sizeof(fortune[0]);

            cout<<"Data yang akan di sort: ";

    Sorting(fortune, size);

            cout<<endl;

            cout<<"Quick Sort: "<<endl;

    QuickSort(fortune, 0, size - 1);


                    cout<<"\n\nMau mengulang program atau
langsung exit?";
                    cout<<"\n\n[1]Mengulang program GAKKOU";
                    cout<<"\n[2]Exit Program";

                    cout<<endl<<endl;

                    cout<<"Pilihan kamu [1/2/?]: ";

                    cin>>pilihan;

                        if(pilihan==1){
```

```
                goto gakkou;

            }

            else if(pilihan==2){

                system("cls");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

                exit(0);

            }

            else{

                system("cls");

                system("COLOR D");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;
```

```
                                        while(kimi_no_na_wa <
tenki_no_ko){

        cout<<"MADAKONOSEKAIWAMINASAGARASUZUMU";

                                                kimi_no_na_wa--;

                                        }


                                }



return 0;

}
```

```
///////Merge Sort kode Script//////

#include <iostream>
#include <string>

using namespace std;



void Sorting(int sort[], int size) {

   for (int i = 0; i < size; i++) {

      cout<<sort[i];

      if(i < size - 1) cout<<", ";

   }

   cout<<"]"<<endl;
}

void merge(int sort[], int low, int mid, int high, int& sekai, int&
yume) {

        int hidarinosize = mid - low + 1;

   int miginosize = high - mid;
```

```
int hidari[hidarinosize];

int migi[miginosize];

for (int i = 0; i < hidarinosize; i++) {

    hidari[i] = sort[low + i];

    yume++;
}

for (int j = 0; j < miginosize; j++) {

    migi[j] = sort[mid + 1 + j];

    yume++;

}

int i=0, j=0, k=low;

while (i < hidarinosize && j < miginosize) {

    sekai++;

    if (hidari[i] >= migi[j]) {
```

```
        sort[k] = hidari[i];

        i++;

    }

            else {

        sort[k] = migi[j];

        j++;
    }

    k++;

    yume++;
}

while (i < hidarinosize) {

    sort[k] = hidari[i];

    i++;

    k++;

    yume++;
```

```
    }

    while (j < miginosize) {

        sort[k] = migi[j];

        j++;

        k++;

        yume++;
    }
}

void MergeSort(int sort[], int low, int high, int& sekai, int& yume) {

    if (low < high) {

        int mid = low + (high - low) / 2;

        cout<<"Proses pemecahan list [";

        Sorting(sort + low, mid - low + 1);

        MergeSort(sort, low, mid, sekai, yume);

        cout<<"Proses pemecahan list [";
```

```
        Sorting(sort + mid + 1, high - mid);

        MergeSort(sort, mid + 1, high, sekai, yume);

        cout<<"Proses penggabungan list [";

        Sorting(sort + low, high - low + 1);

        merge(sort, low, mid, high, sekai, yume);
    }
}



int main(){

system("COLOR F");

        printf("\n\n====================TUGAS PERTEMUAN
2====================\n\n");

                string nma("RIFANDY ARNAS");
                string nim("232310001");
                string kls("TI-23-PA ");

                kls.append("(LAB 2)");
```

```
        string mtkl("Design dan Analisis Algoritma");
        string tgl("22 Maret 2024 ");

        tgl.append("(Pembuatan Script)");

            cout<<"Nama\t: "<<nma;
            cout<<"\nNIM\t: "<<nim;
            cout<<"\nKelas\t: "<<kls;
            cout<<"\n\nMatkul\t: "<<mtkl;
            cout<<"\nTanggal\t: "<<tgl;

    printf("\n\n>>>>>>>>>>>>>>>>>>>>Merge
Sort<<<<<<<<<<<<<<<<<<<\n\n");

        int pilihan; int kimi_no_na_wa; int tenki_no_ko;

        kimi_no_na_wa=1;

        tenki_no_ko=2;

            cout<<"List program kali ini: ";
            cout<<"\n\n[1]SPARKLE";
            cout<<"\n[2]Exit Program";

            cout<<endl<<endl;

            cout<<"Pilihan kamu [1/2/?]: ";
```

```
cin>>pilihan;

if(pilihan==1){

goto sparkle;

}

else if(pilihan==2){

system("cls");


cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

exit(0);


}

else{

system("cls");

system("COLOR D");
```

```
        cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

        cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;


                                        while(kimi_no_na_wa <
tenki_no_ko){

        cout<<"MADAKONOSEKAIWAMINASAGARASUZUMU";

                                                kimi_no_na_wa--;

                                        }



                                }

sparkle:

        system("cls");

        system("COLOR B");

        printf("\n\n//////////////////////SPARKLE//////////////////////\n\n");

                        cout<<"Selamat datang di program SPARKLE!\n";
```

```
                    cout<<"Program ini akan menyortir angka dengan
metode Merge Sort!\n";
                    cout<<"Selamat menikmati!";

                    cout<<endl<<endl<<endl;



            int sayonara[] = {2, 5, 60, 43, 27, 10, 89, 17};

        int size = sizeof(sayonara) / sizeof(sayonara[0]);

        int sekai=0, yume=0;

            cout<<"Data awal: [";

   Sorting(sayonara, size);

   MergeSort(sayonara, 0, size - 1, sekai, yume);

            cout<<"\nData terurut secara descending: [";

   Sorting(sayonara, size);



            cout<<"\n\nMau mengulang program atau langsung
exit?";
                    cout<<"\n\n[1]Mengulang program SPARKLE";
                    cout<<"\n[2]Exit Program";
```

```
cout<<endl<<endl;

cout<<"Pilihan kamu [1/2/?]: ";

cin>>pilihan;

    if(pilihan==1){

        goto sparkle;

    }

    else if(pilihan==2){

        system("cls");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

        exit(0);

    }

    else{
```

```
                                system("cls");

                                system("COLOR D");


        cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

        cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl;


                                        while(kimi_no_na_wa <
tenki_no_ko){

        cout<<"MADAKONOSEKAIWAMINASAGARASUZUMU";

                                                kimi_no_na_wa--;

                                        }


                        }


return 0;
}
```