UTS STRUKTUR DATA

Aldi Maulana Iqbal (20210801222)

1. Buatlah sebuah program Struktur Data Sorting untuk pengurutan data Dengan metode Bubble Sort Asending & Quick sort Desending.

   Fungsi *Bubble Sort Ascending*:

```
// Bubble Sort Ascending Function
void bubbleSwapping(int &a, int &b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}

void bubbleSortAsc(int *array, int size)
{
    for (int i = 0; i < size; i++)
    {
            int swaps = 0;
            for (int j = 0; j < size - i - 1; j++)
            {
                    if (array[j] > array[j + 1])
                    {
                            bubbleSwapping(array[j], array[j + 1]);
                            swaps = 1;
                    }
            }
            if (!swaps)
            {
                    break;
            }
    }
}
```

   Fungsi *Quick Sort Descending*:

```
// Quick Sort Descending Function
void swap(int *array, int left, int right)
{
    int temp = array[left];
    array[left] = array[right];
    array[right] = temp;
}

int partition(int *array, int left, int right)
{
    int pivot = array[right];
    int leftPointer = left - 1;
    int rightPointer = right;

    for (;;)
    {
            while (array[++leftPointer] > pivot)
            {
            }
            while (rightPointer > 0 && array[--rightPointer] < pivot)
            {
            }
            if (leftPointer >= rightPointer)
            {
```

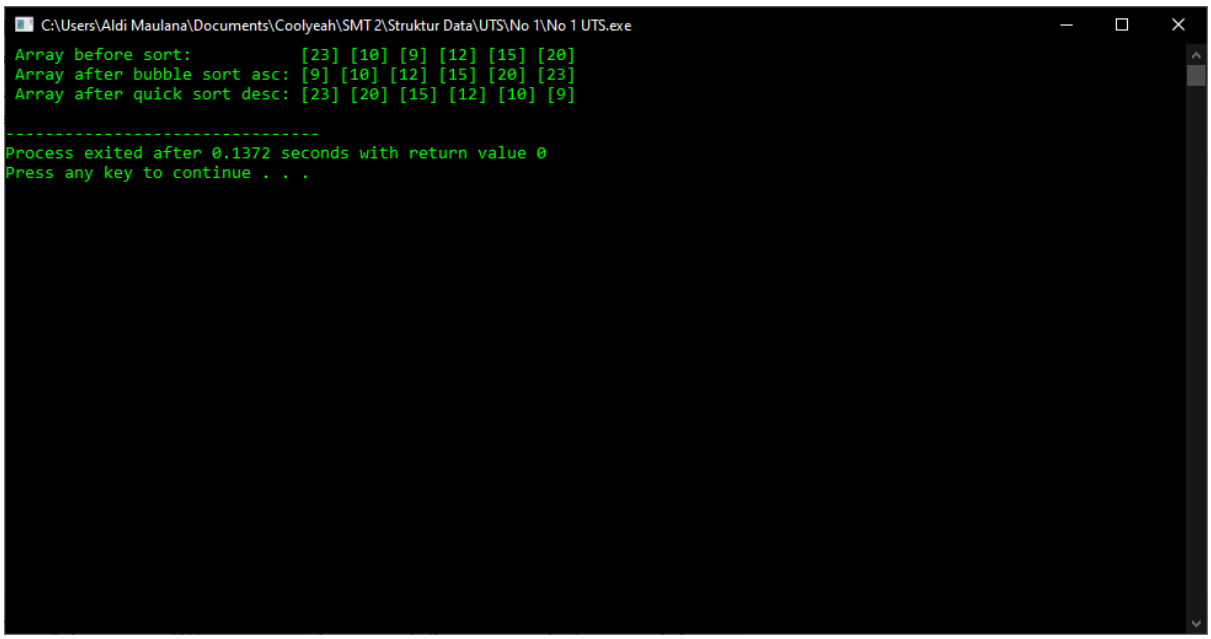
```
                break;
            }
            else
            {
                swap(array, leftPointer, rightPointer);
            }
    }

    swap(array, leftPointer, right);
    return leftPointer;
}

void quicksort(int *array, int left, int right)
{
    if (left < right)
    {
            int partitionPoint = partition(array, left, right);
            quicksort(array, left, partitionPoint - 1);
            quicksort(array, partitionPoint + 1, right);
    }
}
```

Main Code:

```
#include <iostream>
#include <stdio.h>
#include <stdlib.h>

using namespace std;

int main()
{

    int array[6] = {23, 10, 9, 12, 15, 20};
    int arrayLeng = sizeof(array) / sizeof(int);

    cout << " Array before sort:           ";
    for (int i = 0; i < arrayLeng; i++)
    {
            cout << "[" << array[i] << "]"
                    << " ";
    }

    cout << endl;

    bubbleSortAsc(array, arrayLeng);
    cout << " Array after bubble sort asc: ";
    for (int i = 0; i < arrayLeng; i++)
    {
            cout << "[" << array[i] << "]"
                    << " ";
    }

    cout << endl;

    quicksort(array, 0, arrayLeng - 1);
    cout << " Array after quick sort desc: ";
    for (int i = 0; i < arrayLeng; i++)
    {
            cout << "[" << array[i] << "]"
                    << " ";
```

```
    }

    cout << endl;
}
```

Hasil:



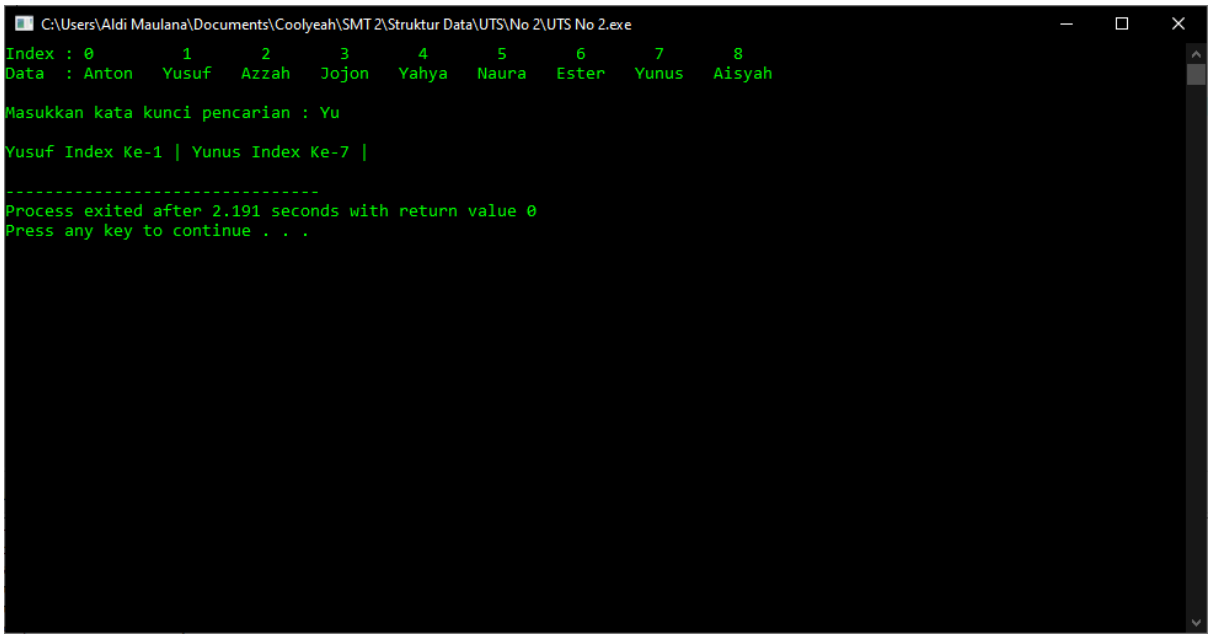
2. Buatlah sebuah program Struktur Data Searching (mencari) data yang huruf nya dimulai dengan Yu* pada Data yang tersimpan dalam Array 1 dimensi berikut ini sampai di temukan semua data yang huruf awalnya Yu*.

   Source Code:

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

using namespace std;

void displayIndex(int size)
{
    for (int i = 0; i < size; i++)
    {
            cout << i << "\t  ";
    }
}

void displayData(string *array, int size)
{
    for (int i = 0; i < size; i++)
    {
            cout << array[i] << "\t";
    }
}

int main()
{
```

```
    string data[9] = {"Anton", "Yusuf", "Azzah", "Jojon", "Yahya",
"Naura", "Ester", "Yunus", "Aisyah"};
    string search;
    int dataLeng = sizeof(data) / sizeof(string);

    cout << "Index : ";
    displayIndex(dataLeng);
    cout << endl;

    cout << "Data  : ";
    displayData(data, dataLeng);
    cout << "\n\n";

    cout << "Masukkan kata kunci pencarian : ";
    cin >> search;

    cout << endl;

    bool found = false;
    for (int i = 0; i < dataLeng; i++)
    {
        if (!data[i].find(search))
        {
            cout << data[i] << " Index Ke-" << i << " | ";
            found = true;
        }
    }

    cout << endl;
    if (!found)
    {
        cout << "Data Not Found! Try Again!";
    }
}
```

Result:

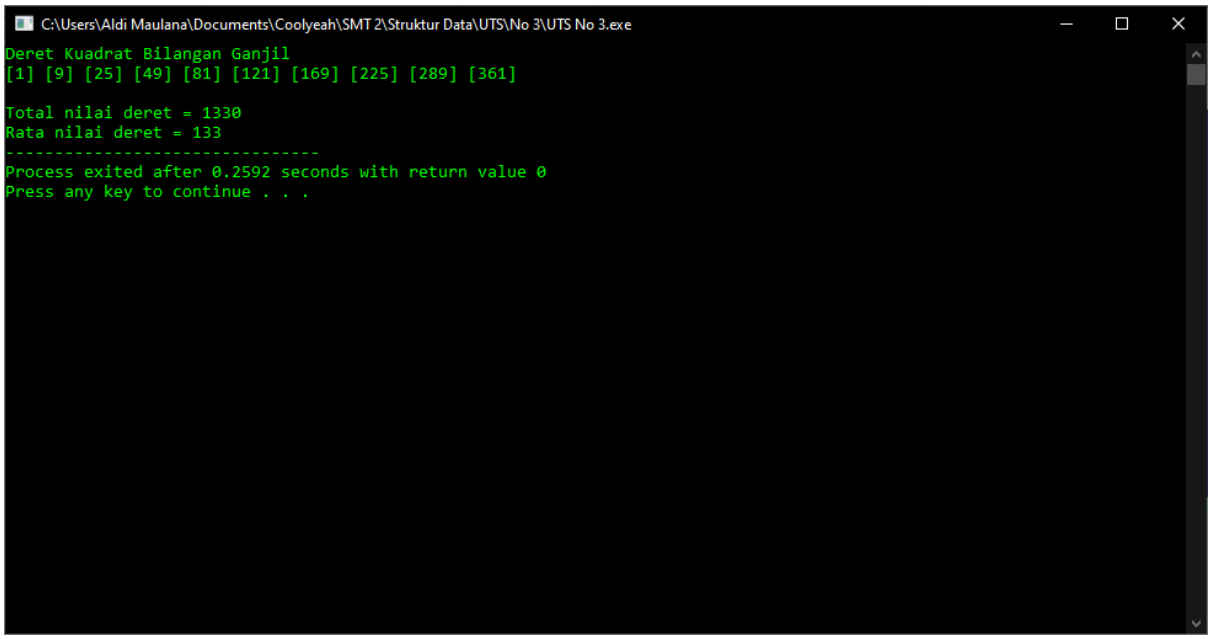


3. Buat script StrukturData + Algoritma dengan salah satu tools (C++, Java, JavaScript, PHP, Python, dan lainya) untuk mencetak Sederetan Data Kuadrat bilangan ganjil seperti dibawah ini, Dimana ada Nilai total Dan nilai rata rata nya.

   Source Code:

```cpp
#include <iostream>
#include <cmath>

using namespace std;

int main()
{

    cout << "Deret Kuadrat Bilangan Ganjil" << endl;
    int num = 1;
    int sum = 0;
    while (num <= 19)
    {
            cout << "[" << num * num << "] ";
            sum += num * num;
            num += 2;
    }
    cout << "\n\n";

    cout << "Total nilai deret = " << sum << endl;
    cout << "Rata nilai deret = " << (float)sum / 10;
}
```

   Result :



4. Bila data Mahasiswa : Nim, Nama, dan IPK yang akan ditampilkan 5 data Mahasiswa, deklarasikanlah struktur Mahasiswa tersebut dgn struktur pointer to array, Buatlah fungsi/prosedure input data dan fungsi atau prosedure tampilan datanya.

Source Code:

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

using namespace std;

struct Mahasiswa
{
    string nama;
    string nim;
    string ipk;
    string keterangan;
};

int main()
{
    char tab = '\t';
    int jmlhMhs = 5;
    int firstCat = 0;
    int secondCat = 0;
    int thirdCat = 0;
    int uncategorizedCat = 0;

    Mahasiswa mhs[jmlhMhs];

    for (int i = 0; i < jmlhMhs; i++)
    {
        cout << "Data-" << i + 1 << " : "
             << "Entry Data" << endl;
        cout << "          Nama   : ";
        getline(cin, mhs[i].nama);
        cout << "          NIM    : ";
        getline(cin, mhs[i].nim);
        cout << "          IPK    : ";
        getline(cin, mhs[i].ipk);

        int ipk = stoi(mhs[i].ipk);

        if (ipk >= 88)
        {
            mhs[i].keterangan = "Sangat Memuaskan";
            firstCat++;
        }
        else if (ipk >= 77 && ipk <= 88)
        {
            mhs[i].keterangan = "Memuaskan";
            secondCat++;
        }
        else if (ipk >= 60 && ipk <= 76)
        {
            mhs[i].keterangan = "Cukup";
            thirdCat++;
        }
        else
        {
            mhs[i].keterangan = "Uncategorized";
            uncategorizedCat++;
        }
    }
```

```
    cout << "                DAFTAR NAMA MHS STRUKTUR DATA                "
<< endl;
    cout << "-------------------------------------------------------------"
<< endl;
    cout << "No.\t\tNama\t\tNIM\t\tIPK\t\tKET." << endl;
    for (int j = 0; j < jmlhMhs; j++)
    {
        cout << j + 1 << tab << tab << mhs[j].nama << tab << tab <<
mhs[j].nim << tab << tab << mhs[j].ipk << tab << tab <<
mhs[j].keterangan << endl;
    }
    cout << endl;
    cout << "Jumlah predikat sangat memuaskan = " << firstCat <<
"\tOrang." << endl;
    cout << "Jumlah predikat memuaskan        = " << secondCat <<
"\tOrang." << endl;
    cout << "Jumlah predikat cukup            = " << thirdCat <<
"\tOrang." << endl;
    cout << "Jumlah predikat uncategorized    = " << uncategorizedCat
<< "\tOrang." << endl;
}
```

Result: