**LAPORAN PRAKTIKUM**

**CODE BLOCKS**

**SORTING**

****

**Disusun Oleh :**

FILFIMO YULFIZ AHSANUL HULQI

NIM : 19102143

**Dosen**

Emi Iryanti, S.ST., M.T

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI DAN INFORMATIKA**

**INSTITUT TEKNOLOGI TELKOM**

**PURWOKERTO**

**2019**

**BAB I**

1. **Tujuan**

Mahasiswa diharapkan mampu :

1. Memahami dan menerapkan beberapa algoritma sorting dalam menyelesaikan berbagai studi kasus.
2. Mahasiswa dapat membuat sebuah program dengan sorting didalamnya
3. Membedakan dan memahami macam-macam algortima sorting dan mana yang palki efektif untuk digunakan
4. **Dasar Teori**

* Sorting adalah algortima untuk meletakkan kumpulan elemen data ke dalam urutan tertentu, berdasarkan satu atau beberapa kunci ke dalam tiap-tiap elemne
* Sorting dibagi menjadi 2 jenis yaitu:
  1. Ascending

Adalah pengurutan dari nilai terkecil ke nilai terbesar, sedangkan,

* 1. Descending

Adalah pengurutan dari nilai terbesar ke nilai terkecil

**BAB II**

Penjelasan Tugas

1. Kelas S1 IF 2016 D memiliki 5 mahasiswa. Pada akhir semester merka menerima lembar Indeks Prestasi Semester (IPS), masing-masing mahasiswa tersebut memiliki IPS sebagai berikut {3.8, 2.9, 3.3, 4.0, 2.4}, Buatlah program untuk mengurutkan IPS mahasiswa tersebut dari yang terbesar hingga terkecil dengan menggunakan algoritmas Selection Sort
2. Pseudo code

#include <iostream>

using namespace std;

void selection\_sort(float arr[], int length){

int tmp, j;

for (int i = 0; i < length - 1; i++){

j = i;

while (j > 0 && arr[j-1] < arr[j]){

tmp = arr[j];

arr[j] = arr[j-1];

arr[j-1] = tmp;

j--;

}

}

}

void print\_array(float a[], int length){

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

cout << a[i] << "\t";

}

cout << endl;

}

int main(){

int length = 5;

float a[length] = {3.8, 2.9, 3.3, 4.0, 2.4};

cout << "IPS Sebelum Di Urutkan" << endl;

print\_array(a, length);

selection\_sort(a, length);

cout << "IPS Sesudah Di Urutkan" << endl;

print\_array(a, length); // Output hasil sorting

return 0;

}

1. Deskripsi kode

void selection\_sort(…..){

Algoritma Selection Sort adalah memilih elemen dengan nilai paling rendah dan menukar elemen yang terpilih dengan elemen ke-i. Nilai dari i dimulai dari 1 ke n, dimana n adalah jumlah total elemen dikurangi 1.

while (j > 0 && arr[j-1] < arr[j]){

tmp = arr[j];

arr[j] = arr[j-1];

arr[j-1] = tmp;

j--;

Nilai dari arr[j], akan disisimpan sementara di tmp, untuk melakukan perbadingan dikurngani dengan -1, terus mengulang sampai porses seleksi selesai.

1. Screenshot Program



1. Pak RT memiliki 10 warga dengan nama Siti, Situ, Sana, Ana, Ani, Caca, Cici, Dida, Dodo, dan Dadi. Supaya mudah dalam melakukan pencarian, Pak RT akan mengurutkan nama-nama tersebut sesuai dengan urutan Alfabet. Buatlah program untuk membantu Pak RT dengan menggunakan algoritma Buuble Sort.
2. Pseudo code

#include <iostream>

using namespace std;

void bubble\_sort(string arr[], int length){

bool not\_sorted = true;

int j = 0;

string tmp;

while (not\_sorted){

not\_sorted = false;

j++;

for (int i = 0; i < length - j; i++){

if (arr[i] > arr[i + 1]){

tmp = arr[i];

arr[i] = arr[i + 1];

arr[i + 1] = tmp;

not\_sorted = true;

}

}

}

void print\_array(string a[], int length){

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

cout << a[i] << "\t";

}

cout << endl;

}

int main()

{

int length = 10;

string a[length] = {"Siti", "Situ", "Sana", "Ana", "Ani", "Caca", "Cici", "Dida", "Dodo", "Dadi"};

cout << "Daftar Nama Sebelum Di Urutkan" << endl;

print\_array(a, length);

cout << endl;

bubble\_sort(a, length);

cout << "Daftar Nama Sesudah Di Urutkan" << endl;

print\_array(a, length);

return 0;

}

1. Deskripsi kode

void bubble\_sort(string arr[], int length){

Algoritma Bubble Sort ini merupakan proses pengurutan yang secara berangsur-angsur berpindah ke posisi yang tepat karena itulah dinamakan Bubble yang artinya gelembung. Algoritma ini akan mengurutkan data dari yang terbesar ke yang terkecil (ascending) atau sebaliknya (descending). Atau bisa didefenisikan Bubble Sort adalah pengurutan dengan cara pertukaran data dengan data disebelahnya secara terus menerus sampai dalam satu iterasi tertentu tidak ada lagi perubahan

for (int i = 0; i < length - j; i++){

if (arr[i] > arr[i + 1]){

tmp = arr[i];

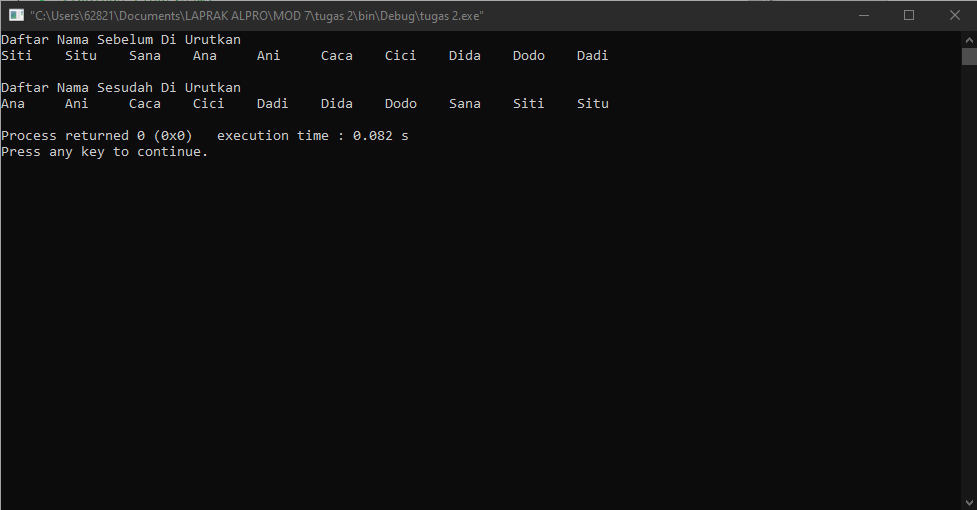
arr[i] = arr[i + 1];

arr[i + 1] = tmp;

not\_sorted = true;

Sama seperti program sebelumnya dimana arr[i] akan disimpan sementara didalam tmp, kemudian akan dibandingkan dengan data-data yang lain, sampai sudah tidak ada lagi perubahan pengurutan.

1. Screesnshot program



1. Buatlah program yang meminta user menginputkan suatu bilangan n dan meminta user untuk menginputkan sejumlah n karakter. Kemudian program akan melakukan sorting secara menaik (ascending) dan menurun (descending)
2. Pseudo Code

#include <iostream>

using namespace std;

void bubble\_sort\_asc(char arr[], int length){

bool not\_sorted = true;

int j = 0;

char tmp;

while (not\_sorted){

not\_sorted = false;

j++;

for (int i = 0; i < length - j; i++){

if (arr[i] > arr[i + 1]){

tmp = arr[i];

arr[i] = arr[i + 1];

arr[i + 1] = tmp;

not\_sorted = true;

}

}

}

}

void bubble\_sort\_des(char arr[], int length){

bool not\_sorted = true;

int j = 0;

char tmp;

while (not\_sorted){

not\_sorted = false;

j++;

for (int i = 0; i < length - j; i++){

if (arr[i] < arr[i + 1]){

tmp = arr[i];

arr[i] = arr[i + 1];

arr[i + 1] = tmp;

not\_sorted = true;

}

}

}

}

void print\_array(char a[], int length){

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

cout << a[i] << "\t";

}

cout << endl;

}

int main(){

cout << "Masukan Nilai N : ";

int length;

cin >> length;

char a[length];

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

cout << "Karakter Ke-" << i+1 << " : ";

cin >> a[i];

}

cout << "Urutan karakter sebelum sorting :" << endl;

print\_array(a, length);

cout << endl;

bubble\_sort\_asc(a, length);

cout << "Urutan karakter sesudah ascending sort :" << endl;

print\_array(a, length);

cout << endl;

bubble\_sort\_des(a, length);

cout << "Urutan karakter sesudah descending sort :" << endl;

print\_array(a, length);

cout << endl;

return 0;

}

1. Deskripsi Kode

Disini menggunakan 2 tipe sorting yaitu ascending dan descending secara bersamaan, dimana juga nanti akan mengouput secara bersama-sama dengan hasil yang berbeda

void bubble\_sort\_asc(char arr[], int length){

bool not\_sorted = true;

Int j = 0;

char tmp;

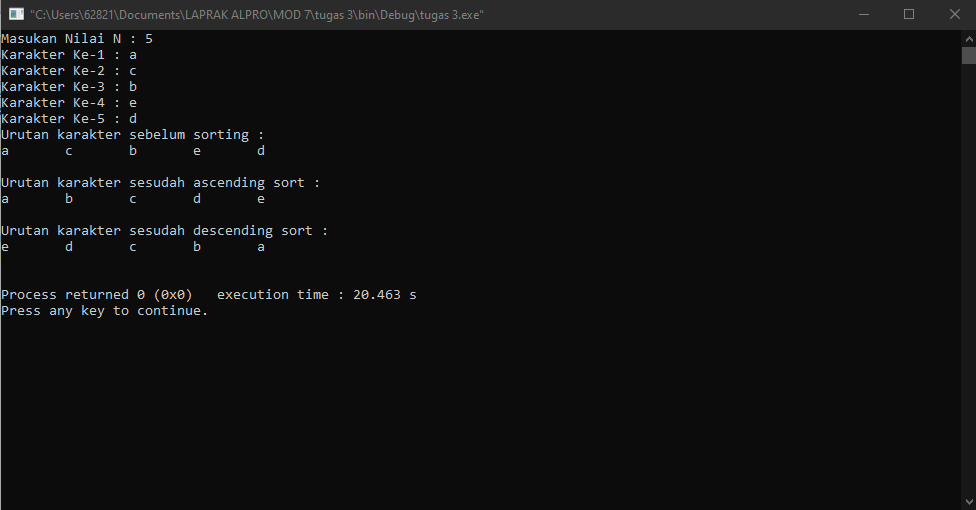
while (not\_sorted){

not\_sorted = false;

j++;……..

Disini prosedur untuk ascending dibuat, dimana akan melakukan penyeleksian data sama seperti tugas 1 dan 2 diatas, dan juga untuk descending juga sama, akan melakukan penyeleksian data

1. Screenshot program



**BAB III**

**Kesimpulan**

Program dapat dijelajahi dan dikembangkan lebih jauh lagi dengan algoritma sorting, Dimana sebelumnya mempelajari sebuah fungsi dan prosedur sederhana, di materi ini mempelajari sebuah prosedur sorting, untuk mengurutkan suatu data yang tidak terurut.

Terdapat banyak jenis Sorting, yang mempunyai karakteristik, dan kecepatan melakukan sorting, ada yang lambat, dan ada juga yang cepat.

Oleh karena itu penggunaan algoritma sorting ini membutuhkan pemahaman tentang jenis sorting supaya penggunaannya efektif dan efisien.

**Daftar Pustaka**

Tekno Jurnal, <https://teknojurnal.com/pengertian-algoritma-bubble-sort/>, diakses pada 12/6/2019.

<http://pendtiumpraz.blogspot.com/2012/03/selection-sort-descending.html>, diakses pada 12/6/2019.