**LAPORAN PRAKTIKUM**

**CODE BLOCKS**

**SEARCHING**

****

**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 algortima pencarian (searching) dalam menyelesaikan berbagai kasus.
2. Mehamahami penggunaan dan dasar-dasar searching.
3. Memahami perbedaan Sequential dan Binary search.
4. **Dasar Teori**
5. Konsep dasar searching

Searching adalah Proses mendapatkan informasi berdasarkan kata kunci tertentu dari sejumlah informasi yang telah disimpan.

1. Jenis-jenis searching
2. Sequential Search

Sequential Search merupakan metode pencarian data dalam array dengan cara membandingkan data yang dicari dengan data yang ada di dalam array secara berurutan.

1. Binary Search

Salah satu syarat pencarian biner (binary search) dapat dilakukan adalah data sudah dalam keadaan terurut. Dengan kata lain, apabila data belum dalam keadaan terurut , pencarian biner tidak dapat dilakukan.

**BAB II**

Penjelasan Tugas

1. Pak Polisi memiliki database plat nomer mobil. Terdapat 10 nomor dalam database tersebut, yaitu sebagai berikut: R 300 SR, R 1234 DJ, R 3218 RR, R 701 LP, R 007 TU, R 3 ST, R 210 RO, R 1111 II, dan R 4987 LH. Pada suatu hari Pak Polisi melihat kendaraan ber-nomer R 999 RS berada di area dilarang parker. Bantulah Pak Polisi untuk mencari apakah nomer tersebut ada di dalam database atau tidak! Gunakan algortima Sequential Search!
2. Source Code

#include <iostream>

using namespace std;

int sequential\_search(string data[], int panjang, string key){

int posisi, ketemu;

if(panjang <= 0){

posisi = -1;

} else {

ketemu = 0;

int i = 1;

while ((i < panjang - 1) && !ketemu){

if (data[i] == key) {

posisi = i;

ketemu = 1;

} else {

i++;

}

if (!ketemu){

posisi = -1;}

}

}

return posisi;}

int main(){

int panjang = 10;

cout << "==========PROGRAM SEARCHING PLAT NOMER==========" << endl <<endl;

string data[panjang]={ "R 300 SR", "R 1234 DJ", "R 3218 RR", "R 701 LP", "R 007 TU", "R 3 ST", "R 999 RT", "R 210 RO", "R 1111 II", "R 4987 LH"}; //Isi dari data

string key = "R 999 RS";

int posisi = sequential\_search(data, panjang, key);

if (posisi != -1){

cout << "Plat " << key << " yang anda cari ditemukan di indeks ke-" << posisi << endl;

} else {

cout << "Plat " << key << " yang anda cari tidak ditemukan" << endl;

}

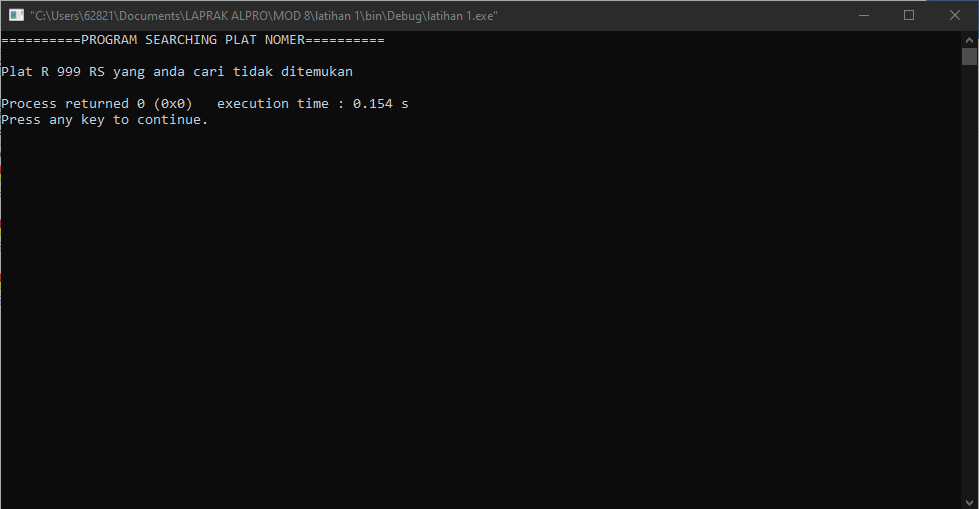
return 0;

}

1. Deskripsi kode

Diatas aada funsgi sequential search, Pencarian ini melakukan pengulangan dari 1 sampai dengan jynlah data, dimana di atas jumlah data bernilai 10 int panjang = 10; , Yang akan dibandingkan dengan key atau kunci yang sedang dicari yaitu R 999 RS.

1. Screenshot Program



1. Dalam satu kelas terdapat 13 mahasiswa yang memiliki nim sebagai berikut: 12102001, 12102002, 12102003, 12102004, 12102005, 12102006, 12102007, 12102008, 12102009, 12102010, 12102011, 12102012, dan 12102013. Dengan menggunakan Algoritma Binary Search, carilah NIM 12102011 apakah berada di kelas tersebut atau tidak.
2. Source code

#include <iostream>

using namespace std;

int binery(int data[], int panjang, int key){

int atas, bawah, tengah, posisi;

bool ada;

ada = false;

bawah = 0;

atas = panjang - 1;

posisi = -1;

while (bawah <= atas){

tengah = (atas + bawah)/2;

if (key == data[tengah]) {

posisi = tengah;

break;

} else if (key < data[tengah]) {

atas = tengah - 1;

} else if (key > data[tengah]) {

bawah = tengah + 1;

}

}

return posisi;}

int main(){

int panjang = 13;

cout << "==========PENCARIAN NIM MAHASISWA==========" << endl << endl;

int data[panjang]={12102001, 12102002, 12102003, 12102004, 12102005, 12102006, 12102007, 12102008, 12102009, 12102010, 12102011, 12102012, 12102013};

int key = 12102011;

int posisi = binery(data, panjang, key);

if (posisi != -1){

cout << "Nim " << key << " yang anda cari ditemukan di indeks ke-" << posisi << endl;

} else {

cout << "Nim " << key << " yang anda cari tidak ditemukan" << endl;

}

return 0;

}

1. Deskripsi kode

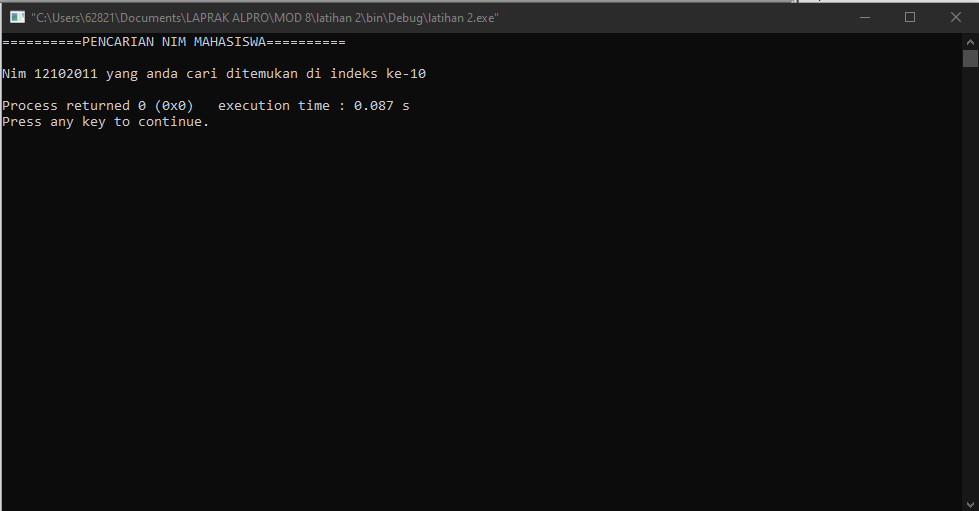
Pada nomor 2, kita menggunakan Binary Search, cara kerja dari Binary search adalah melakukan pembagian pada data yang tersedia. Tetapi untuk menggunakan Binary Search ada 1 syarat yang harus terpenuhi yaitu, Data harus terurut, jika data belum terurut maka binary search tidak dapat dilakukan

int atas, bawah, tengah, posisi; pertama dideklariskan atas/awal, bawah/akhir, tengah dan posisi,

while (bawah <= atas){

tengah = (atas + bawah)/2; Seperti yang disinggung diatas bahwa searching ini akan membagi data, antara kiri dan kanan, melakukan pengecekan apakah data tersebut lebih besar atau lebih kecil, kemudian melakukan perbandingan lagi, hingga menemukan data sesuai key atau tidak ditemukan data yang tidak dicaari

1. Screenshot program

****

1. Pak Anto membuat program untuk men-generate bilangan acak. Saat program dijalankan, program memberikan daftar bilangan acak sebagai berikut: 21, 61, 28, 72, 44, 68, 37, 52, 75, dan 75. Bantulah Pak Anto membuat program pencarian untuk bilangan acak tersebut dengan menggunakan Algoritma Binary Search. Angka yang dicari adalah 71.
2. Source code

#include <iostream>

using namespace std;

int binary(int data[], int panjang, int key){

int atas, bawah, tengah, posisi;

bool ada;

ada = false;

bawah = 0;

atas = panjang - 1;

posisi = -1;

while (bawah <= atas){

tengah = (atas + bawah)/2;

if (key == data[tengah]) {

posisi = tengah;

break;

} else if (key < data[tengah]) {

atas = tengah - 1;

} else if (key > data[tengah]) {

bawah = tengah + 1;

}

} return posisi;}

int main(){

int panjang = 10;

int data[panjang]={21, 61, 28, 72, 44, 68, 37, 52, 75, 75};

int key = 71;

int posisi = binary(data, panjang, key);

if (posisi != -1){

cout << "Angka " << key << " yang anda cari ditemukan di indeks ke-" << posisi << endl;

} else {

cout << "Angka " << key << " yang anda cari tidak ditemukan" << endl;

}

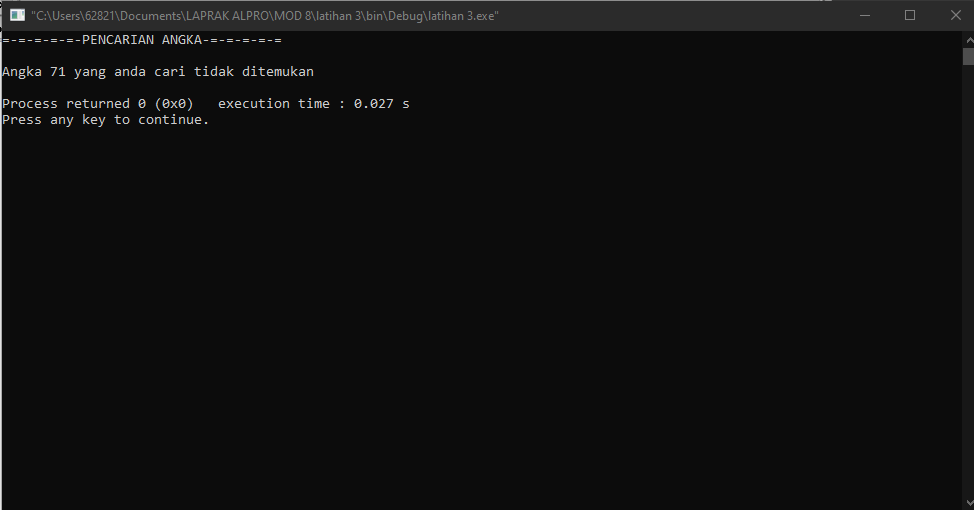
return 0;

}

1. Deskripsi kode

Sama seperti nomer 2 diatas, disini menggunakan Binary Search. Disini kita mengurutkan data terlebih dahulu sesuai syarat yang telah disebutkan diatas. Menjadi : 21, 28, 37, 44, 52, 61, 68, 72, 75, 75.

1. Screenshot Program



**BAB III**

**Kesimpulan**

Program dapat dijelajahi lebih lanjut dan dikembangkan lebih baik dengan Algortima Searching, Dimana di materi ini kita mengenal jenis-jeni searching dan bagaimana menggunakan keduanya, searching dapat dikolaborasikan dengan berbaagai algortima lain terutama Sorting, jika kita akan menggunakan Binary Search, yang mana akan sangat membantu.

Terdapat 2 jenis Searching, yang mempunayi karakteristik, yang berbeda.

Oleh karena itu penggunaan searching ini membutuhkan pemahaman sorting dan searching supaya penggunaannya lebih efektif.\

**Daftar Pustaka**

<http://guntur98.blogspot.com/2015/08/pengertian-dan-jenis-jenis-searching.html>, diakses pada 12/13/2019.

<http://anantyaade.blogspot.com/2017/07/fungsi-pencarian-binary-search-dan.html>, diakses pada 12/13/2019.

<http://onlyvista.blogspot.com/2017/07/pengertian-searching-jenis-jenis.html>, diakses pada 12/13/2019.