**Laporan Hasil Ujian Tengah Semester (UTS) Pemikiran Komputasional**

**“Sistem Manajemen Pengelolaan Data Mahasiswa Berbasis CLI”**



Disusun Oleh:

1. Muhammad Refdi Pasaribu / 231110267
2. Muhammad Siddik Syahputra / 231111876
3. Ahmad Yuda Zaki Yamani / 231112933
4. Nurihsan Febrianto / 231111759

**TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**UNIVERSITAS MIKROSKIL**

**2024**

# Daftar Isi

[1. Daftar Isi 2](#_Toc165984179)

[2. Algoritma Sistem Aplikasi CRUD Data Mahasiswa Menggunakan Bahasa Pemrograman Python 3](#_Toc165984180)

[3. Kompleksitas Program 5](#_Toc165984181)

[3.1 Tambah Data Mahasiswa O(1) – Konstan 5](#_Toc165984182)

[3.2 Tampilkan Data Mahasiswa O(1) - Konstan 5](#_Toc165984183)

[3.3 Ubah Data Mahasiswa O(1) – Konstan 6](#_Toc165984184)

[3.4 Pengurutan Data Mahasiswa O(n2) 6](#_Toc165984185)

[3.5 Pencarian Data Mahasiswa O(log n) 7](#_Toc165984186)

[4. Alur Sistem Aplikasi CRUD Data Mahasiswa Menggunakan Bahasa Pemograman Pyhton 9](#_Toc165984187)

[4.1 Main menu 9](#_Toc165984188)

[4.2 First menu (Create Student Data) 9](#_Toc165984189)

[4.3 Second menu(Read Student Data) 9](#_Toc165984190)

[4.4 Third menu(Update Student Data) 10](#_Toc165984191)

[4.5 Fourth menu(Sorting Student Data by Field) 11](#_Toc165984192)

[4.6 Fifth menu(Search Student Data by NIM, Name) 12](#_Toc165984193)

[4.7 Last menu(Exit) 14](#_Toc165984194)

[5. Pembagian Tugas Anggota 15](#_Toc165984195)

# Algoritma Sistem Aplikasi CRUD Data Mahasiswa Menggunakan Bahasa Pemrograman Python

1. Mulai
2. Masukkan perintah yang ingin dijalankan 1-6 : int
   1. Tambah Data Mahasiswa (1)
      1. Masukkan *`NIM`* Mahasiswa
      2. Masukkan *`Nama`* Mahasiswa
      3. Masukkan *`Tempat Lahir`* Mahasiswa
      4. Masukkan *`Tanggal Lahir`* Mahasiswa
      5. Masukkan *`Program Studi`* Mahasiswa
      6. Masukkan *`Tahun Masuk`* Mahasiswa
   2. Tampilkan Data Mahasiswa (2)
      1. Semua data mahasiswa akan ditampilkan dalam bentuk tabel dengan format {nim, nama, tempat\_tanggal\_lahir, program\_studi, tahun\_masuk}
   3. Ubah Data Mahasiswa (3)
      1. Masukkan *`NIM`* Mahasiswa yang ingin di ubah datanya
      2. Data mahasiswa akan di tampilkan dalam bentuk tabel dengan format {field, detail}
      3. Masukkan *`Nama`* mahasiswa jika ingin mengubahnya
      4. Masukkan *`Tempat Lahir`* mahasiswa jika ingin mengubahnya
      5. Masukkan *`Tanggal Lahir`* mahasiswa jika ingin mengubahnya
      6. Masukkan *`Program Studi`* mahasiswa jika ingin mengubahnya
      7. Masukkan *`Tahun Masuk`* mahasiswa jika ingin mengubahnya
   4. Urutkan Data Mahasiswa (4)
      1. Masukkan pilihan urutan berdasarkan field (1-4) : int
         1. `NIM`
         2. `Nama`
         3. `Program Studi`
         4. `Tahun Masuk`
      2. Setelah memilih salah satu ke-empat menu tersebut data akan di tampilkan secara (ascending) sesuai field yang di pilih, format untuk tabel yang akan ditampilkan {nim, nama, tempat\_tanggal\_lahir, program\_studi, tahun\_masuk}
   5. Cari Data Mahasiswa (5)
      1. Masukkan pilihan pencarian berdasarkan (1-2) : int
         1. `NIM`
            1. Masukkan *`NIM`* Mahasiswa yang ingin dicari
            2. Jika *`NIM`* dari mahasiswa ketemu maka

Tampilkan dalam bentuk tabel dengan format {nim, nama, tempat\_tanggal\_lahir, program\_studi, tahun\_masuk}

* + - * 1. Jika tidak maka

Tampilkan pesan kesalahan *“Data tidak ditemukan.”*

* + - 1. `Nama`
         1. Masukkan *`Nama`* Mahasiswa yang ingin dicari
         2. Jika *`Nama`* dari mahasiswa ketemu maka

Tampilkan dalam bentuk tabel dengan format {nim, nama, tempat\_tanggal\_lahir, program\_studi, tahun\_masuk}

* + - * 1. Jika tidak maka

Tampilkan pesan kesalahan *“Data tidak ditemukan.”*

* 1. Keluar (6)
     1. Program akan dihentikan

1. Selesai

# Kompleksitas Program

## Tambah Data Mahasiswa O(1) – Konstan



Gambar 3.1

Kompleksitasnya bisa dikatakan O(1), karena jumlah operasi yang dilakukan tidak bergantung pada jumlah masukan yang sudah ada dalam aplikasi. Jadi, tidak peduli seberapa banyak data yang telah ada, proses penambahan data baru ini akan memiliki kompleksitas yang relatif konstan.

## Tampilkan Data Mahasiswa O(1) - Konstan



Gambar 3.2

Kompleksitas ini juga bisa dikatakan O(1), karena jumlah operasi hanya dilakukan satu kali.

## Ubah Data Mahasiswa O(1) – Konstan



Gambar 3.3

Kompleksitasnya bisa dikatakan O(1), karena jumlah operasi yang dilakukan tidak bergantung pada jumlah masukan yang sudah ada dalam aplikasi. Jadi, tidak peduli seberapa banyak data yang telah ada, proses perubahan data ini akan memiliki kompleksitas yang relatif konstan.

## Pengurutan Data Mahasiswa O(n2)



Gambar 3.4

Kompleksitas waktu dari algoritma quicksort O(n^2),karena pivot dipilih selalu merupakan elemen terbesar atau terkecil dalam array. Jadi, jika list sudah terurut maka algoritma ini akan melakukan iterasi dari elemen yang lebih besar.

## Pencarian Data Mahasiswa O(log n)



Gambar 3.5

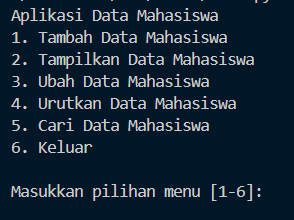
Kompleksitas dari Binary searchnya adalah O(log n), namun di fungsi `sort\_data` adalah O(n2), karena Binary search lebih dominan digunakan makanya kompleksitas pada fungsi `search\_data` ini adalah O(log n)

# Alur Sistem Aplikasi CRUD Data Mahasiswa Menggunakan Bahasa Pemograman Pyhton

## Main menu

Description:

When run, the program immediately presents an interactive menu that makes it easy for users to use this data CRUD application.

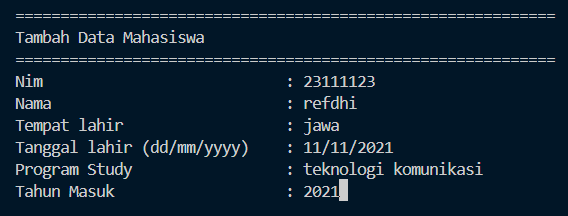


Gambar 4.1

## First menu (Create Student Data)

Description:

In this first menu, users can enter the desired student data in a structured manner.



Gambar 4.2

## Second menu(Read Student Data)

Descriptiom:

This second menu will display the student data table that has been inputted from the first menu.

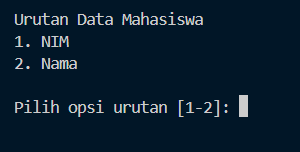


Gambar 4.3

## Third menu(Update Student Data)

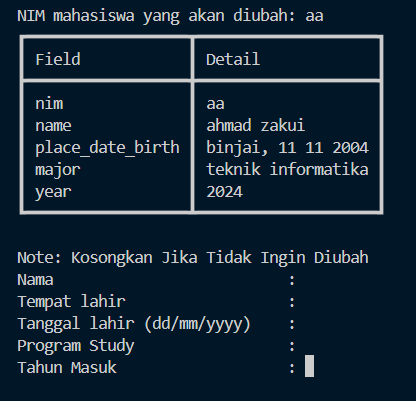
Description:

on this menu, allows users to change data that already exists in the table, based on name or nim.



Gambar 4.4

If selecting NIM then, the user must input the appropriate nim in the previous data and change the data as desired. Likewise, if selecting NAMA.

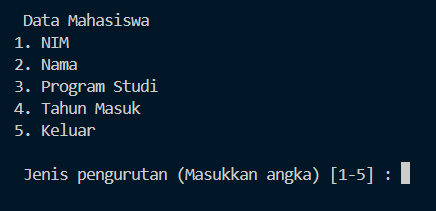


Gambar 4.5

## Fourth menu(Sorting Student Data by Field)

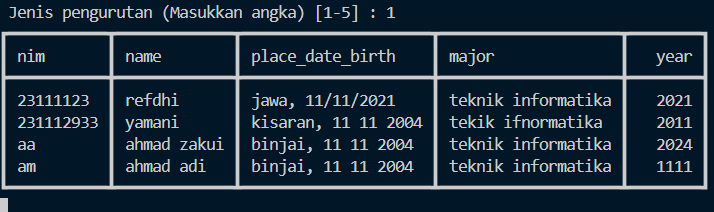
Description:

in the fourth menu the user will be given the option to sort the data according to the options provided.



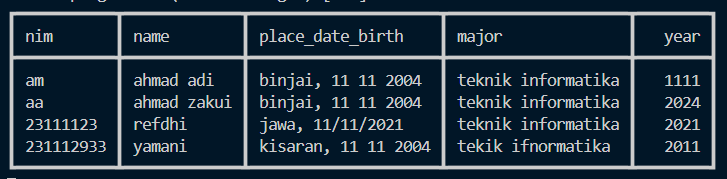
Gambar 4.6

if the user selects the NIM option, the data will be sorted by NIM.



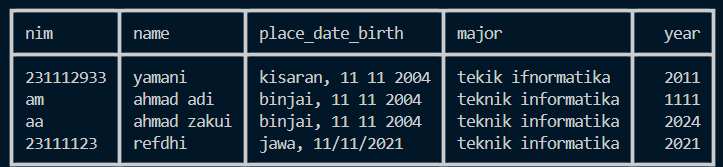
Gambar 4.7

if the user selects the Nama option, the data will be sorted by Nama.



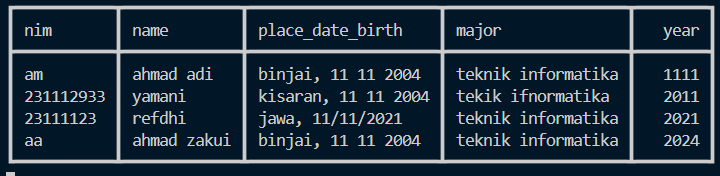
Gambar 4.8

if the user selects the Program Studi option, the data will be sorted by Program Studi.



Gambar 4.9

if the user selects the Tahun Masuk option, the data will be sorted by Tahun Masuk.



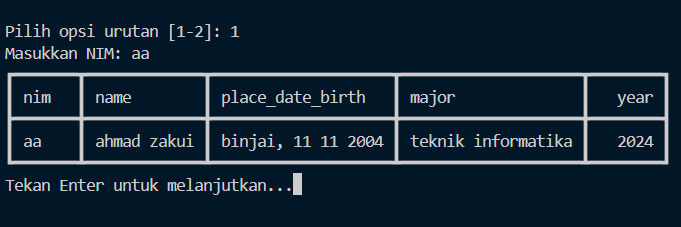
Gambar 4.10

## Fifth menu(Search Student Data by NIM, Name)

Description:

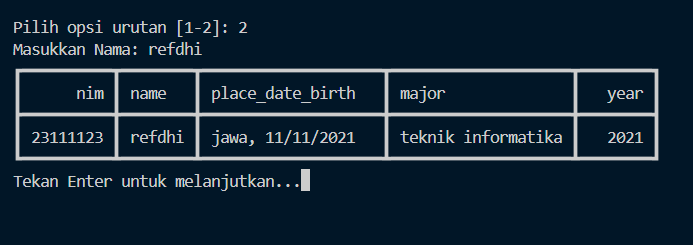
in this menu the user is allowed to search for the desired student data with the NIM and Name options.

if you choose the NIM option, then the student data search will be based on the NIM inputted by the user and if the input is correct, it will display the student data.



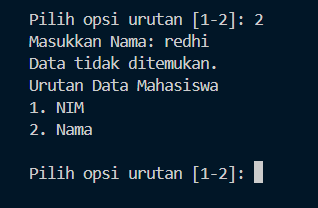
Gambar 4.11

if you choose the Nama option, then the student data search will be based on the Nama inputted by the user and if the input is correct, it will display the student data.



Gambar 4.12

if the user inputs data that does not match then no data is displayed and the user will be asked to re-input.



Gambar 4.13

## Last menu(Exit)

to quit or exit the application

# Pembagian Tugas Anggota

1. Muhammad Refdi Pasaribu
   1. Merancang alur dan penggunaan library pada sistem aplikasi
   2. Menentukan kompleksitas program
   3. Membuat Laporan
   4. Membuat Algoritma
2. Muhammad Siddik Syahputra
   1. Merancang alur dan penggunaan library pada system aplikasi
   2. Menentukan kompleksitas program
   3. Membuat Laporan
   4. Membuat Alur Program
3. Ahmad Yuda Zaki Yamani
   1. Setup projek aplikasi hasil dari rancangan yang sudah didiskusikan
   2. Implementasi dari fungsi fungsi yang sudah dirancang
   3. Maintance sistem aplikasi termasuk perbaikan bug
   4. Membuat Laporan
4. Nurihsan Febrianto
   1. Merancang struktur data json untuk penampungan data
   2. Merancang semua fungsi untuk pengelolahan data
   3. Membuat Laporan
   4. Testing aplikasi, memastikan aplikasi sudah berjalan dengan benar dan effisien