**SISTEM INFORMASI AKADEMIK (SIAKAD) : PROGRAM PENGELOLA DATA AKADEMIK MAHASISWA**

LAPORAN STUDI KASUS TUGAS AOL

MATA KULIAH COMP6360004 – ALGORITHM AND PROGRAMMING

KELAS LB-20



Oleh :

2602080636 – KEVIN FARRELL FEBRIANUS TANAMATRAS

Semester Ganjil 2022/2023

MALANG

# DAFTAR ISI

[DAFTAR ISI 1](#_Toc125632691)

[BAB I PENDAHULUAN 2](#_Toc125632692)

[BAB II USULAN DESAIN PROGRAM 3](#_Toc125632693)

[BAB III PROGRAM DAN PENJELASANNYA 4](#_Toc125632694)

[BAB IV TAMPILAN HASIL 11](#_Toc125632695)

[BAB V KESIMPULAN 15](#_Toc125632696)

# BAB I

**PENDAHULUAN**

Teknologi informasi, komputer, elektronik, dan automasi ditemukan pada abad ke-20 yang ditandai dengan era revolusi industri 3.0. Di Indonesia sendiri, teknologi ini masuk pada akhir tahun 1960-an dan berkembang pesat hingga saat ini. Komputerisasi adalah proses sistematis oleh komputer berdasarkan instruksi program yang diberikan, sehingga mampu menggantikan proses pengerjaan secara manual dan konvensional. Kecanggihan ini membawa banyak kemudahan bagi masyarakat Indonesia di berbagai sektor, salah satunya di sektor pendidikan. Instansi pendidikan, khususnya perguruan tinggi, membutuhkan teknologi informasi untuk menghimpun ratusan hingga ribuan data dalam waktu yang singkat. Dengan demikian, kegiatan administratif kampus akan berjalan lebih efisien dan efektif. Maka dari itu, diperlukan suatu program bernama Sistem Informasi Akademik (SIAKAD).

Sistem Informasi Akademik (SIAKAD) adalah sebuah program dalam format C yang menerapkan fungsi *sorting, search, input, display*, dan *delete*. SIAKAD digunakan untuk mengelola data demi menunjang kegiatan akademik mahasiswa. Data yang dimaksud dalam hal ini adalah Nomor Induk Mahasiswa (NIM), kelas, nilai, Grade Point Average (GPA), dan nama mahasiswa. Program ini dapat diakses oleh mahasiswa, dosen, dan tenaga pendidik kampus. Berikut ini adalah beberapa fitur yang terdapat dalam SIAKAD:

1. Memasukkan, menyimpan, dan menghapus data mahasiswa.
2. Mencari dan menampilkan data mahasiswa yang dipilih.
3. Mengurutkan data mahasiswa berdasarkan NIM, kelas, nilai, atau nama.

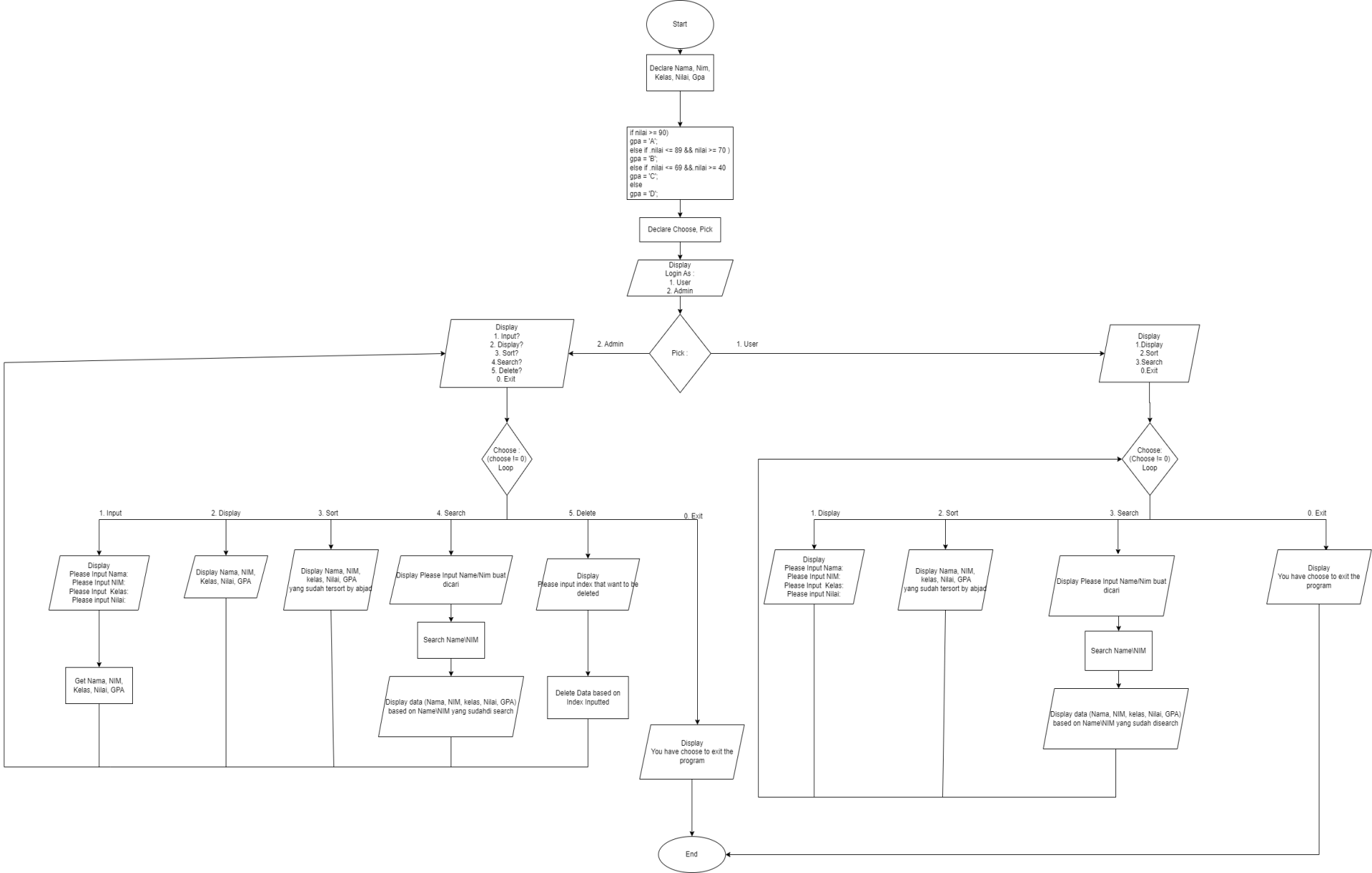
Di tampilan awal, Sistem Informasi Akademik (SIAKAD) akan mengidentifikasi apakah masuk sebagai user atau sebagai admin. Perbedaan mendasar dari user dan admin terletak pada fitur *input* dan *delete*. Bila masuk sebagai admin, pengguna dapat memasukkan, menyimpan, dan menghapus data mahasiswa. Namun, bila masuk sebagai user, pengguna tidak dapat melakukannya. Lebih dari itu, user dan admin memiliki akses yang sama untuk fitur *display*, *sort*, *search*, dan *exit*. Sistem yang dibuat ini bersifat luring/*offline*, artinya tidak terhubung dengan internet.

Data yang masuk akan diklasifikasikan menjadi beberapa tipe data berbeda. Tipe data pertama yaitu integer idx sebagai variabel global dan indeks bagi masing-masing variabel. Selanjutnya, tipe data string untuk variabel nama, kelas, dan NIM. Tipe data integer untuk variabel nilai. Serta tipe data char untuk variabel gpa. Struct yang berisi data (nama, NIM, kelas, nilai, dan GPA) akan tersimpan dalam sebuah file bernama “siakad” dengan format txt. Karena bersifat luring, pengguna hanya bisa membuka program apabila sudah mengunduh file-nya terlebih dahulu.

Dibandingkan dengan cara konvensional, SIAKAD mampu meminimalisir kesalahan data (penghitungan nilai GPA, pengurutan data, pencarian data, dsb.) serta mengurangi risiko terjadinya kehilangan data. Harapannya, Sistem Informasi Akademik (SIAKAD) dapat membantu perguruan tinggi dalam mengelola administrasi kampus secara lebih efektif dan efisien.

# BAB II

**USULAN DESAIN PROGRAM**

****

# BAB III

**PROGRAM DAN PENJELASANNYA**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <stdbool.h>

struct siakad{

char name[100]; //variabel nama yang nantinya akan digunakan untuk menginputkan nama mahasiswa

char nim[100]; //variabel nimyang nantinya akan digunakan untuk menginputkan nim mahasiswa

char kelas[100]; //variabel kelas yang nantinya akan digunakan untuk menginputkan kelas mahasiswa

int nilai;//variabel nilai yang nantinya akan digunakan untuk menginputkan nilai mahasiswa

char gpa; //variabel gpa yang nantinya akan digunakan untuk menginputkan gpa mahasiswa

}siakad[100];

int idx; //global variabel yang nantinya digunakan sebagai indeks dari setiap variabel

void entry() { //fungsi entry untuk menginput data

FILE \*fp;

fp = fopen("siakad.txt", "a"); //membuka file dengan nama siakad

printf("\nSilahkan Masukkan Nama Mahasiswa: "); //display silahkan masukkan nama mahasiswa

scanf("%[^\n]", siakad[idx].name); getchar(); //memasukkan string yang diakhiri dengan \n ke dalam variabel nama di struct siakad dengan indeks ke idx

printf("\nSilahkan Masukkan NIM: "); //display silahkan masukkan nim:

scanf("%[^\n]", siakad[idx].nim); getchar(); //memasukkan string yang diakhiri dengan \n ke dalam variabel nim di struct siakad dengan indeks ke idx

printf("\nKelas: "); //display kelas:

scanf("%[^\n]", siakad[idx].kelas); getchar(); //memasukkan string yang diakhiri dengan \n ke dalam variabel kelas di struct siakad dengan indeks ke idx

printf("\nInput Nilai Rata2 Mahasiswa: "); //display Input nilai rata2 mahasiswa

scanf("%d", &siakad[idx].nilai); getchar(); //memasukkan string yang diakhiri dengan \n ke dalam variabel nilai di struct siakad dengan indeks ke idx

if (siakad[idx].nilai >= 90) { //jika nilai siakad dengan indeks ke idx didalam struct siakad bernilai lebih atau sama dengan 90 maka variabel gpa dengan indeks ke idx yang sama di dalam struct siakad berisi A

siakad[idx].gpa = 'A';

}

else if (siakad[idx].nilai <= 89 && siakad[idx].nilai >= 70 ) { //jika nilai siakad dengan indeks ke idx didalam struct siakad bernilai lebih atau sama dengan 70 dan kurang dari sama dengan 89 maka variabel gpa dengan indeks ke idx yang sama di dalam struct siakad berisi B

siakad[idx].gpa = 'B';

}

else if (siakad[idx].nilai <= 69 && siakad[idx].nilai >= 40 ) { //jika nilai siakad dengan indeks ke idx didalam struct siakad bernilai lebih atau sama dengan 40 dan kurang dari sama dengan 69 maka variabel gpa dengan indeks ke idx yang sama di dalam struct siakad berisi C

siakad[idx].gpa = 'C';

}

else { //jika variabel nilai dalam struct siakad dengan indeks ke idx kurang dari 40 maka variabel gpa dalam struct siakad dengan indeks ke idx berisi D

siakad[idx].gpa = 'D';

}

fprintf(fp, "%s,%s,%d,%c,%s\n", siakad[idx].name, siakad[idx].kelas, siakad[idx].nilai, siakad[idx].gpa, siakad[idx].nim); //printf masing2 variabel kedalam file siakad dengan masing2 format

fclose(fp); //tutup file agar tidak terjadi error

idx++; //lalu increment indeksnya idx agar siap untuk menyimpan value dari masing2 variabel berikutnya

}

void sortAndDisplay() { //fungsi sortanddisplay

int i, j;

struct siakad temp; //membuat struct siakad dengan nama temp

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

for (j = 0; j < idx - i - 1; j++) { //looping untuk mensorting

if (strcoll(siakad[j].name, siakad[j + 1].name) > 0) { //strcoll untuk membandingkan 2 string nama dengan indeks ke j dan selanjutnya yaitu j + 1 jika benar

temp = siakad[j]; // maka value dari variabel temp akan dipindah menjadi value dari variabel siakad dengan indeks ke j

siakad[j] = siakad[j + 1]; // dan value dari siakad dengan indeks ke j akan dipindah menjadi value dari siakad dengan indeks selanjutnya yaitu j+1

siakad[j + 1] = temp; //value dari siakad dengan indeks j+1 akan dipindah menjadi value dari variabel temp

}

}

}

printf("Nama:\t\tKelas:\tNilai:\tGPA:\tNIM:\n"); //printf format display yang berisi nama kelas nilai gpa dan nim diatas lalu

for (int i = 0; i < idx; i++){ // looping i

if (strlen(siakad[i].name) >= 8) //jika panjang string name yang ada di siakad dengan indeks ke i sama tau lebih dari 8 maka lakukan yang ada di statement if jika tidak lakukan yang ada di statement else

{

printf("%s\t%s\t%d\t%c\t%s\n", siakad[i].name, siakad[i].kelas, siakad[i].nilai, siakad[i].gpa, siakad[i].nim);} //printf masing2 variabel dengan indeks ke i

else {

printf("%s\t\t%s\t%d\t%c\t%s\n", siakad[i].name, siakad[i].kelas, siakad[i].nilai, siakad[i].gpa, siakad[i].nim); } //printf masing2 variabel dengan indeks ke i

}

}

void display() { //fungsi display yang berguna untuk menampilkan menu display

printf("Nama:\t\tKelas:\tNilai:\tGPA:\tNIM:\n"); //printf format display yang berisi nama kelas nilai gpa dan nim diatas lalu

for (int i = 0; i < idx; i++) //looping i

{

if (strlen(siakad[i].name) >= 8) //jika panjang string name yang ada di siakad dengan indeks ke i sama tau lebih dari 8 maka lakukan yang ada di statement if jika tidak lakukan yang ada di statement else

{

printf("%s\t%s\t%d\t%c\t%s\n", siakad[i].name, siakad[i].kelas, siakad[i].nilai, siakad[i].gpa, siakad[i].nim);//printf masing2 variabel dengan indeks ke i

}

else {

printf("%s\t\t%s\t%d\t%c\t%s\n", siakad[i].name, siakad[i].kelas, siakad[i].nilai, siakad[i].gpa, siakad[i].nim);//printf masing2 variabel dengan indeks ke i

}

}

}

void deletedata() {//fungsi deletedata yang digunakan untuk menghapus data yang sudah diinput

int x; //declare variabel x

FILE \*fp;

fp = fopen("siakad.txt", "w"); //membuka file siakad yang nantinya akan di w(write)

printf("Select index data you want to delete: ");

scanf("%d", &x); //display selext index data you want to delete dan nantinya apapun yang diinput akan dimasukkan ke dalam variabel x

for (int i = x-1; i < idx; i++) //looping

{

siakad[i] = siakad[i+1]; //value dari siakad dengan indeks ke i akan berpindah menjadi value dari variabel siakad selanjutnya yaitu i+1

}

idx--; //lalu di decrement untuk menghilangkan data yang sudah dipindah tadi

for (int i = 0; i < idx; i++) //looping ke i

{

fprintf(fp, "%s,%s,%d,%c,%s\n", siakad[i].name, siakad[i].kelas, siakad[i].nilai, siakad[i].gpa, siakad[i].nim);//lalu menuliskan kembali ke dalam file dengan masing2 variabel

}

fclose(fp); //lalu menutup file agar tidak terjadi error

}

void search(){//fungsi search yang berguna untuk mensearch apa yang diinput user

char cari[100]; //declare string cari

printf("Nama mahasiswa: ");// display nama mahasiswa:

scanf("%[^\n]", cari);//yang diinput diakhiri dengan \n akan dimasukkan kedalam variabel cari

bool exist = false; //deklarasi variabel bernama "exist" yang bertipe data boolean (true/false) dan diinisialisasi dengan nilai "false".

system("cls"); //perintah untuk membersihkan layar

for (int i = 0; i < idx; i++) //looping i

{

if (strcmp(cari, siakad[i].name) == 0 || strcmp(cari, siakad[i].nim) == 0) { //strcmp digunakan untuk mengcompare variabel name dan variabel cari lalu strcmp lagi untuk mengcompare variabel cari dan nim jika sama stringnya maka menjalankan apa yang ada di dalam bracket

printf("Nama mahasiswa\t: %s\n", siakad[i].name); //printf apa yang ada di variabel name di struct siakad dengan indeks ke i

printf("NIM\t\t: %s\n", siakad[i].nim); //printf apa yang ada di variabel nim di struct siakad dengan indeks ke i

printf("Kelas\t\t: %s\n", siakad[i].kelas); //printf apa yang ada di variabel kelas di struct siakad dengan indeks ke i

printf("Nilai\t\t: %d\n", siakad[i].nilai); //printf apa yang ada di variabel nilai di struct siakad dengan indeks ke i

printf("GPA\t\t: %c\n", siakad[i].gpa); //printf apa yang ada di variabel gpa di struct siakad dengan indeks ke i

exist = true; //mengubah nilai exist menjadi true

system("pause"); //menjalankan perintah pause

}

}

if(exist == false) { //jika setelah looping selesai exist masih bernilai false maka lakukan apa yang ada di dalam bracket

printf("System : item doesn't exist!\n"); //display item doesnt exist

printf("System : returning to main menu...\n");//display returning to main menu

system("pause");//jalankan perintah pause

system("cls");//lalu bersihkan layar

}

}

int main()

{

idx = 0; //declare dan set idx menjadi 0

FILE \*fp;

fp = fopen("siakad.txt", "r"); //buka file siakad untuk dibaca

while (fscanf(fp, "%[^,],%[^,],%d,%c,%s\r\n", siakad[idx].name, siakad[idx].kelas, &siakad[idx].nilai, &siakad[idx].gpa, siakad[idx].nim) != EOF) {

idx++; //jika ketika discan file formatnya "%[^,],%[^,],%d,%c,%s\r\n" dan isinya tidak sama dengan End of file maka tambahkan value dari idx

}

fclose(fp); //tutup lagi file

int Choose, pick; //declare choose dan pick untuk menjadi switch case program

printf("===================================================\n");

printf("================ SIAKAD MAHASISWA =================\n"); //display judul program diatas

printf("===================================================\n");

printf("Login as?\n1.User\n2.Admin\n"); //display login as admin or user

scanf("%d", &pick); //value yang diinput akan dimasukkan kedalam variabel pick

system("cls"); //lalu bersihkan layar

do { //lakukan perintah yang ada didalam bracket selama while terpenuhi yaitu selama yang diinput tidak 0

printf("\n===================================================\n");

printf("============= SIAKAD MAHASISWA ====================\n"); //display lagi judul program diatas

printf("===================================================\n");

if (pick == 1){ //jika pick valuenya 1 maka akan masuk ke menu user

printf("=============------------------====================\n");

printf("=============-- WELCOME USER --====================\n"); //display welcome user untuk menunjukkan bahwa kita masuk sebagai user

printf("=============------------------====================\n");

printf("\n1. Display?\n2. Sort(berdasarkan nama Mahasiswa)?\n3. Cari Mahasiswa(dari nama/nim)?\n0. to exit program\n\nChoose : "); //display menu2 yang tersedia sebagai user

scanf("%d", &Choose); fflush(stdin); //dan apapun yang dimasukkan akan diinput ke dalam variabel choose dan fflush untuk mengflush input buffer

switch (Choose) { //switchcase dan yang digunakan adalah variabel choose tadi

case 1:{

system("cls"); //jika choose valuenya 1 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi display lalu break

display();

break;

}

case 2:{ //jika choose valuenya 2 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi sortanddisplay lalu break

system("cls");

sortAndDisplay();

break;

}

case 3:{ //jika choose valuenya 3 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi search lalu break

system("cls");

search();

break;

}

case 0: { //jika choose = 0 maka break

break;

}

default: { //dan jika memasukkan diluar 1,2,3, dan 0 maka display please input the correct number in order to use the program!

printf("Please Input the correct number in order to use the program!\n");

break;

}

}

}

else if (pick == 2) { //jika choose

printf("============--------------------===================\n");

printf("============---- ADMIN MODE ----===================\n");//display ADMIN MODE untuk menunjukkan bahwa kita masuk sebagai admin

printf("============--------------------===================\n");

printf("\n1. Input data mahasiswa?\n2. Display data mahasiswa?\n3. Sort(berdasarkan nama Mahasiswa)?\n4. Cari Mahasiswa(dari nama/nim)?\n5. Delete data mahasiswa?\n0. to exit program\n\nChoose : "); //display menu2 yang tersedia sebagai admin

scanf("%d", &Choose); fflush(stdin); //dan apapun yang dimasukkan akan diinput ke dalam variabel choose dan fflush untuk mengflush input buffer

switch (Choose) { //switchcase dan yang digunakan adalah variabel choose tadi

case 1: { //jika choose valuenya 1 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi entry lalu jalankan perintah pause dan bersihkan layar

system("cls");

entry();

system("pause");

system("cls");

break;

}

case 2:{ //jika choose valuenya 2 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi display

system("cls");

display();

break;

}

case 3:{ //jika choose valuenya 3 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi sortanddisplay

system("cls");

sortAndDisplay();

break;

}

case 4:{ //jika choose valuenya 4 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi search

system("cls");

search();

break;

}

case 5:{ //jika choose valuenya 5 maka bersihkan layar dulu lalu lanjut untuk menjalankan fungsi display dan deletedata

system("cls");

display();

deletedata();

break;

}

case 0: { //jika choose valuenya 0 maka break loop

break;

}

default: { //jika choose valuenya bukan 1,2,3,4,5 dan 0 maka display please input the correct number in order to use the program

printf("Please Input the correct number in order to use the program!\n");

break;

}

}

}

} while (Choose != 0); //semua program do akan terus dijalankan selama choose tidak sama dengan 0 dan akan berakhir jika choose 0 dan display you have close the program jika choose valuenya 0

printf("\nYou have close the program...");

return 0;

}

# BAB IV

**TAMPILAN HASIL**

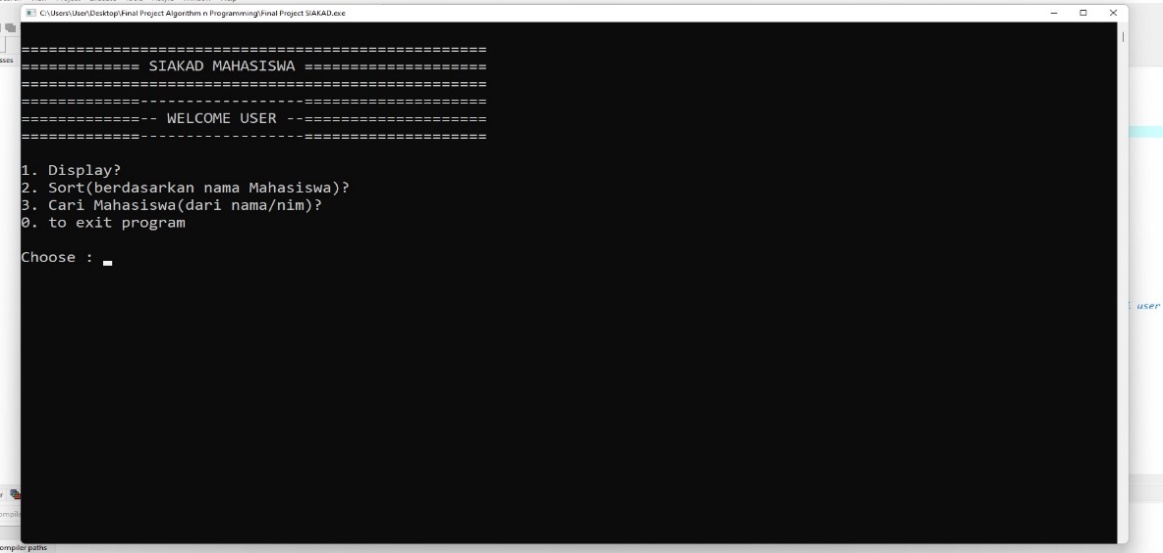
1. Tampilan awal SIAKAD

**Sebuah gambar berisi teks

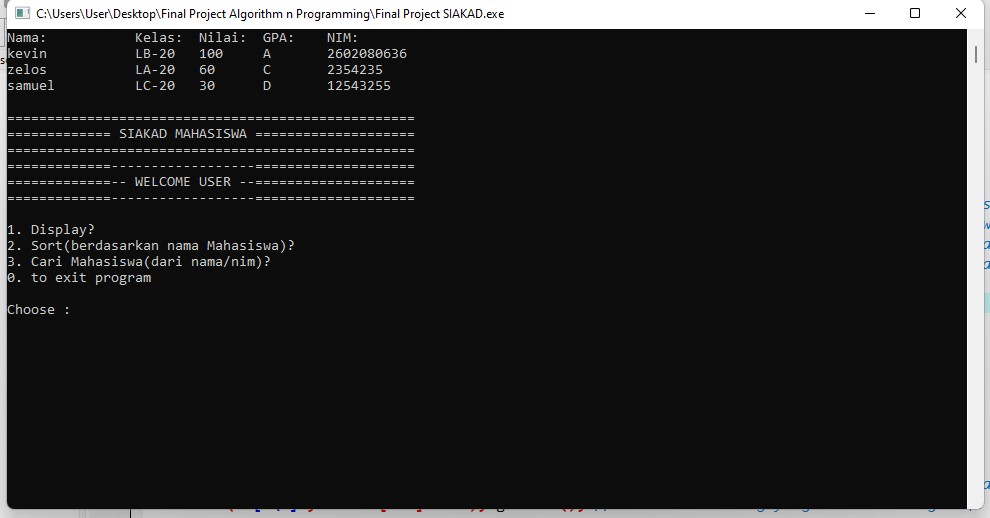
Description automatically generated**

**USER**

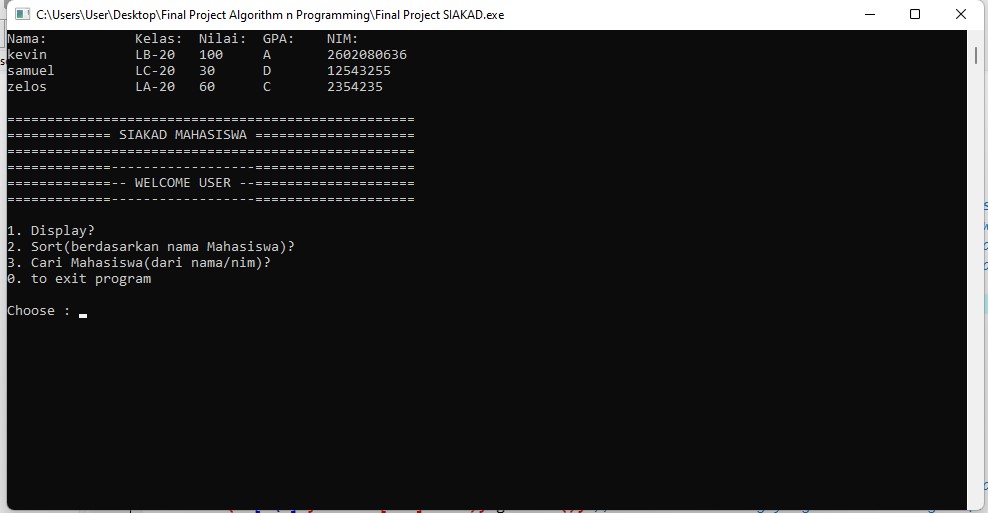
1. Menu user

****

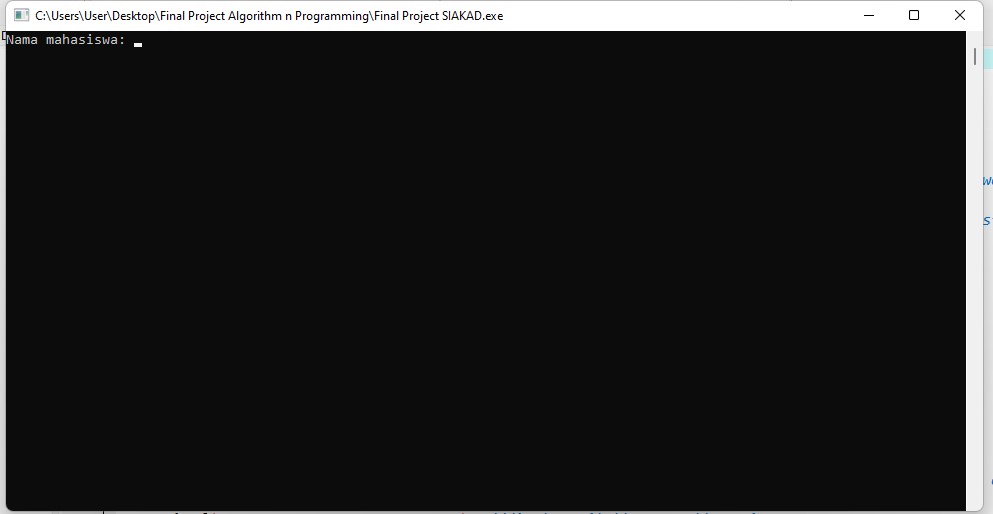
1. Hasil display



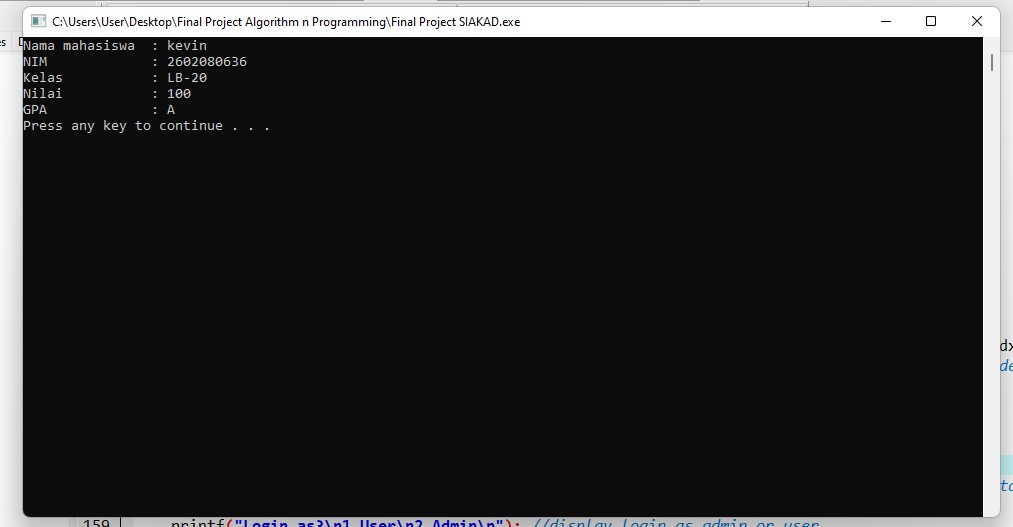
1. Hasil sort



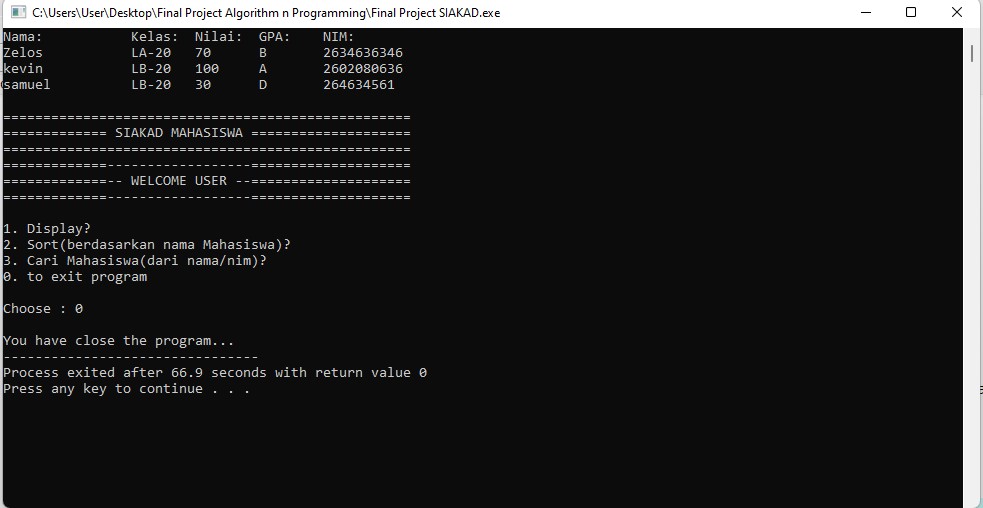
1. Search

****

1. Hasil search

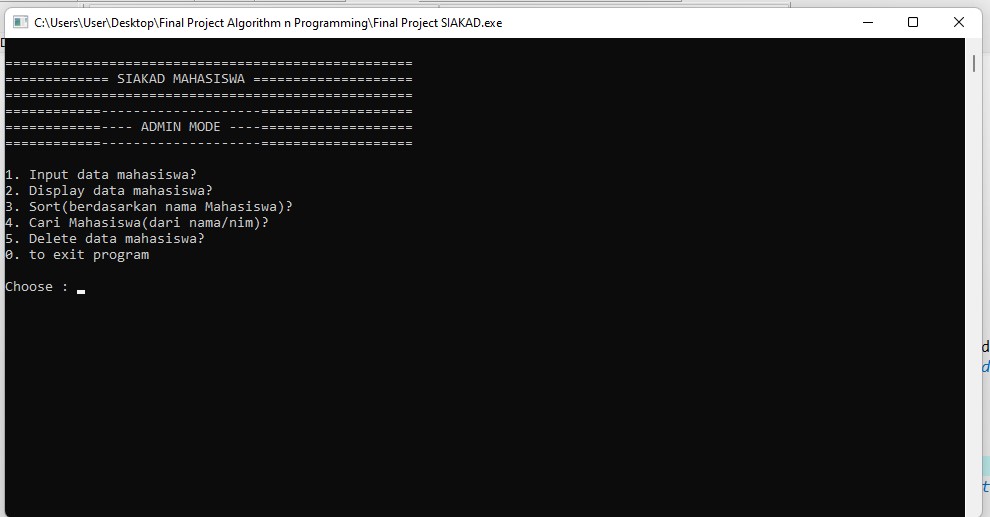
****

1. Exit program

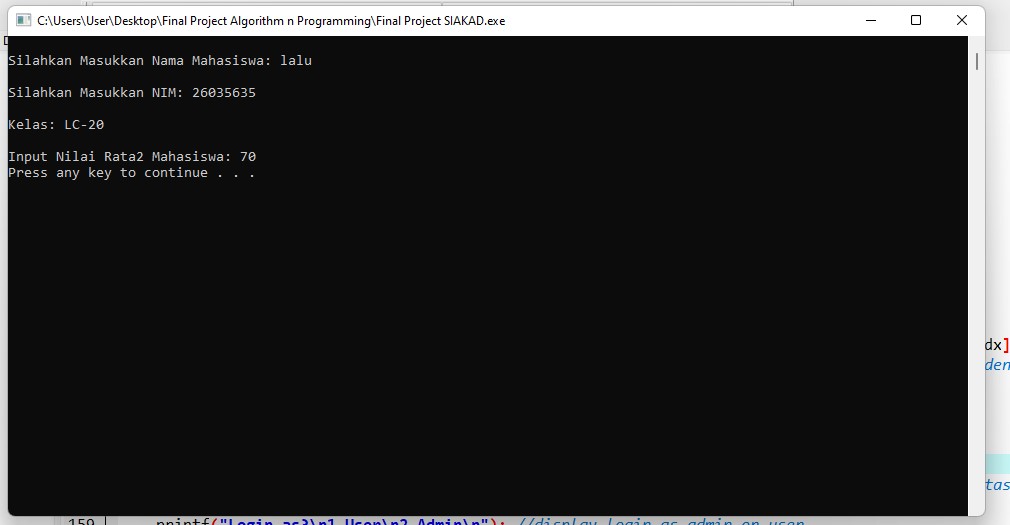
****

**ADMIN**

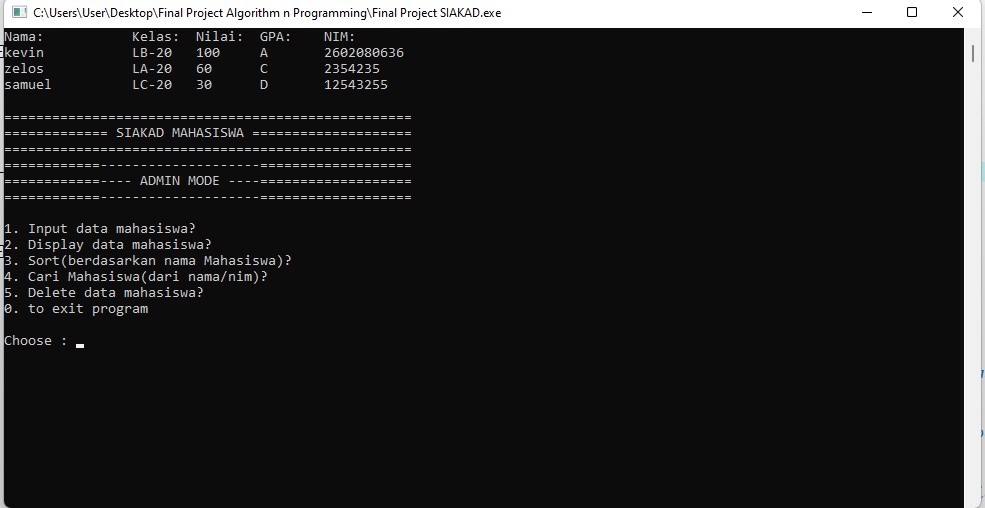
1. Menu

****

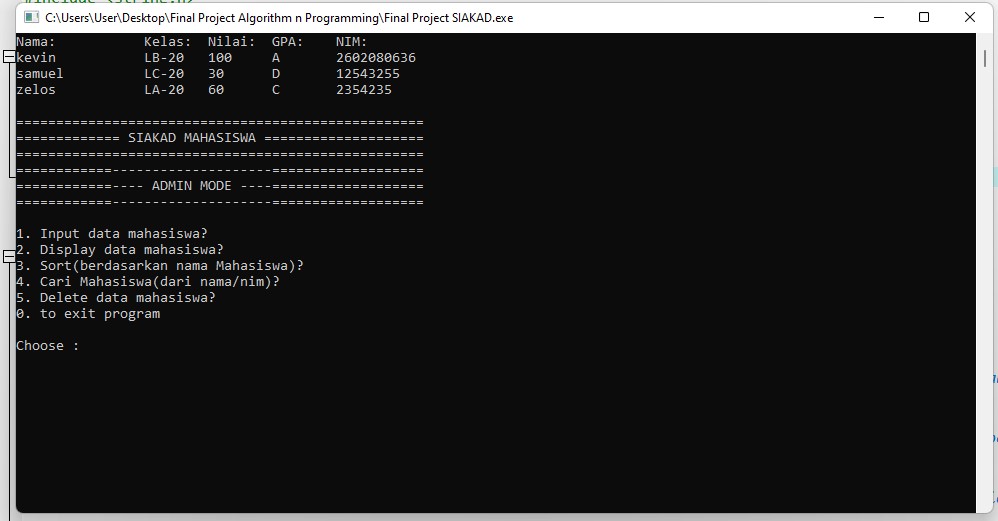
1. Hasil input

****

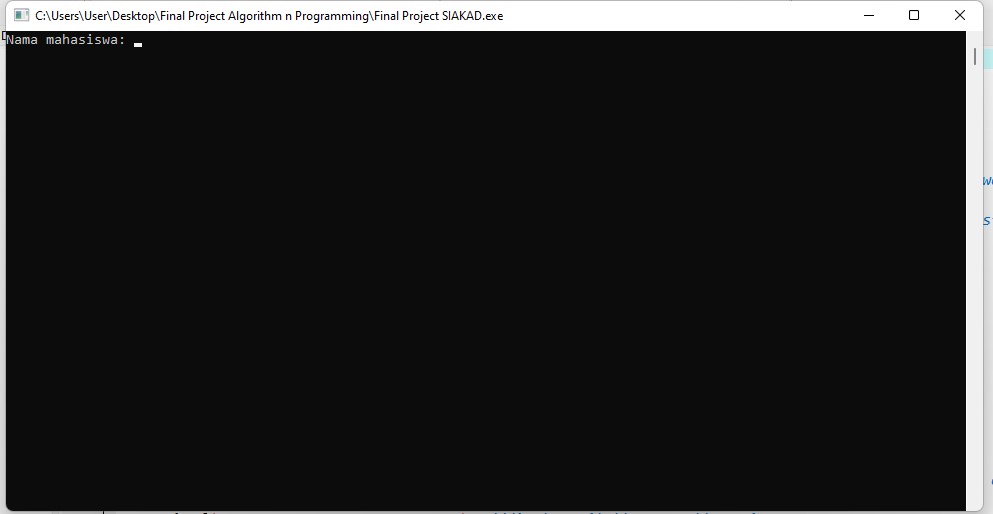
1. Hasil display



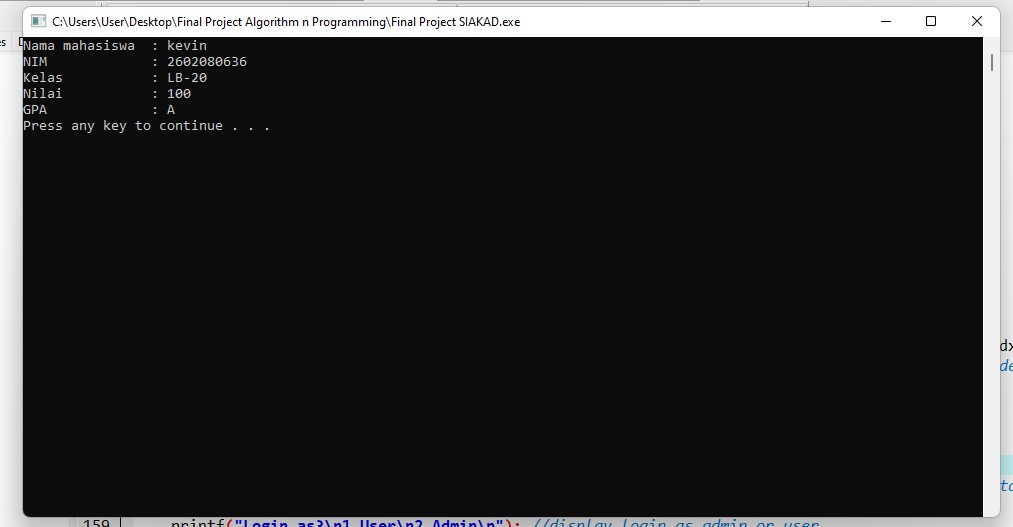
1. Hasil sort

****

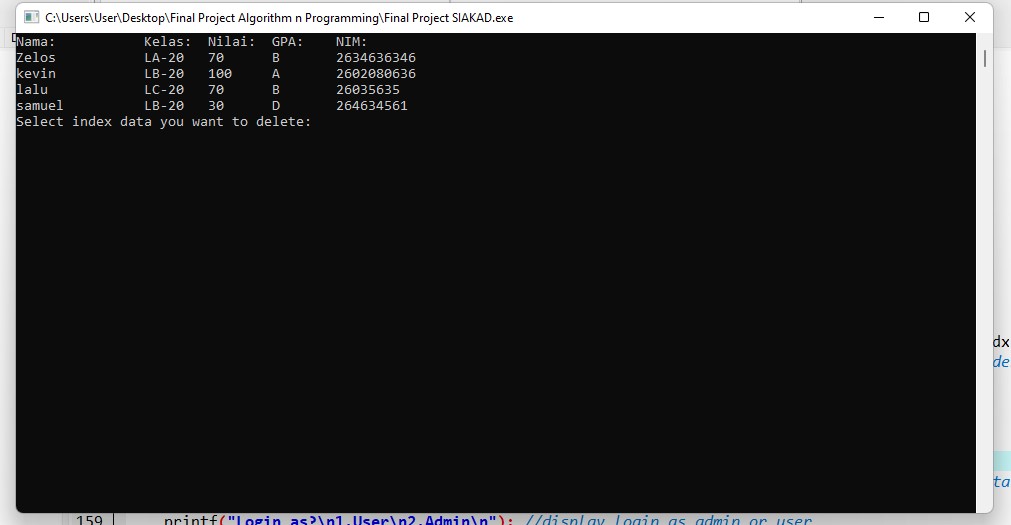
1. Search

****

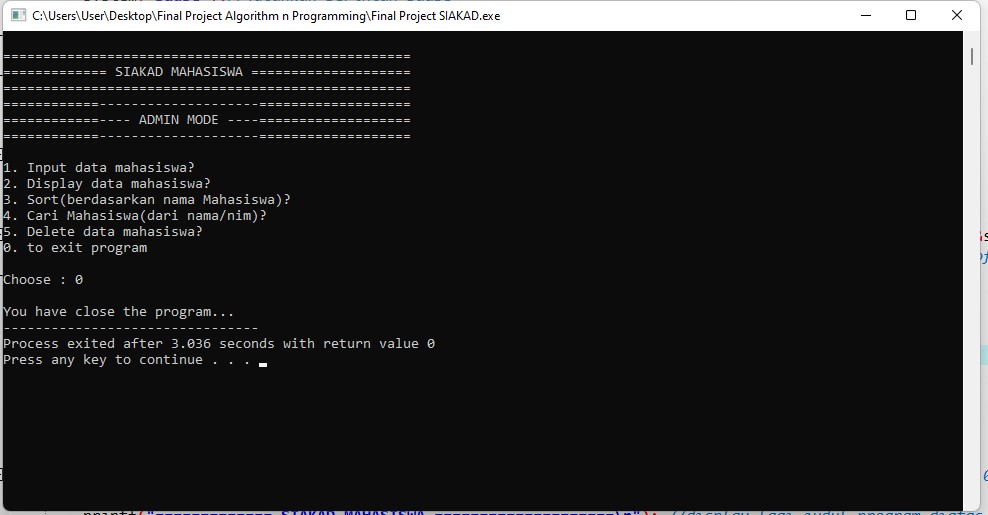
1. Hasil search

****

1. Hasil delete

****

1. Hasil exit program

****

# BAB V

**KESIMPULAN**

Sistem Informasi Akademik (SIAKAD) adalah sebuah program dalam format C yang menerapkan fungsi *sorting, search, input, display*, dan *delete* untuk mengelola data mahasiswa. Data yang tertera berupa Nomor Induk Mahasiswa (NIM), kelas, nilai, nama, dan *Grade Point Average* (GPA). Data tersebut disimpan dalam file “siakad.txt”. Ketika program dijalankan, SIAKAD akan mengidentifikasi status pengguna apakah sebagai user atau admin. Perbedaan mendasarnya terletak pada fitur *input* dan *delete*. Lebih dari itu, keduanya mampu mengakses fitur lain yaitu: *sort* (mengurutkan data), *search* (mencari data mahasiswa), *display* (menampilkan data), serta *exit* untuk keluar dari program. SIAKAD diharapkan mampu membantu perguruan tinggi dalam mengelola data mahasiswa secara lebih efektif dan efisien. Hingga akhirnya, program ini dapat menunjang kegiatan akademik mahasiswa selama berkuliah.