**IDEAL BODY WEIGHT CALCULATOR**

LAPORAN STUDI KASUS TUGAS AOL

MATA KULIAH COMP6360004 – ALGORITHM AND PROGRAMMING KELAS LA20

Oleh :

2602076866 – Albertus Giovanni Prasetyo

Semester Ganjil 2022/2023

MALANG

Daftar Isi

Bab 1 Pendahuluan 3

Bab 2 Usulan Desain Program 4-13

Bab 3 Program dan Penjelasannya 14-21

Bab 4 Tampilan Hasil 22-26

Bab 5 Kesimpulan 28

Bab 1

Pendahuluan

Ide Project – Ideal Body Weight Calculator

Ideal Body Weight Calculator (dapat disingkat IBWCalc) adalah sebuah program yang didesain untuk mengetahui apakah berat badan seseorang sudah ideal atau apakah berat badan orang tersebut kurang/melebihi batasan berat badan ideal. Program ini dibuat dengan menggunakan bahasa pemrograman C. Berikut ini adalah beberapa unsur yang digunakan saat membuat program IBWCalc:

File Processing

File Processing digunakan untuk memasukkan data, membaca data, mengurutkan data, mencari data, mengubah data dan menghapus data yang telah dimasukkan oleh user saat menjalankan program.

Media Penyimpanan

Media penyimpanan yang digunakan untuk menyimpan data yang telah di input oleh user berupa file txt yang bernama “bbideal.txt” yang dipakai untuk melakukan semua kegiatan file processing. Ada juga media sementara (temp file) yang berupa file txt bernama “bbideal2.txt” yang dipakai untuk menampung data sementara sebelum diubah dan dipindahkan ke “bbideal.txt” lalu akan dihapus sebelum program berakhir.

Tipe Data

“bbideal.txt” dan “bbideal2.txt” menyimpan beberapa data berupa nama dengan tipe data string, tinggi dan berat badan dengan tipe data int, bmi (body mass index) dan batas berat badan ideal dengan tipe data float.

Selection

Program IBWCalc ini memiliki beberapa menu, dimana selection dapat membantu user untuk memilih menu apa yang akan digunakan.

Bab 2

Usulan Desain Program

Fungsi untuk mencari nama

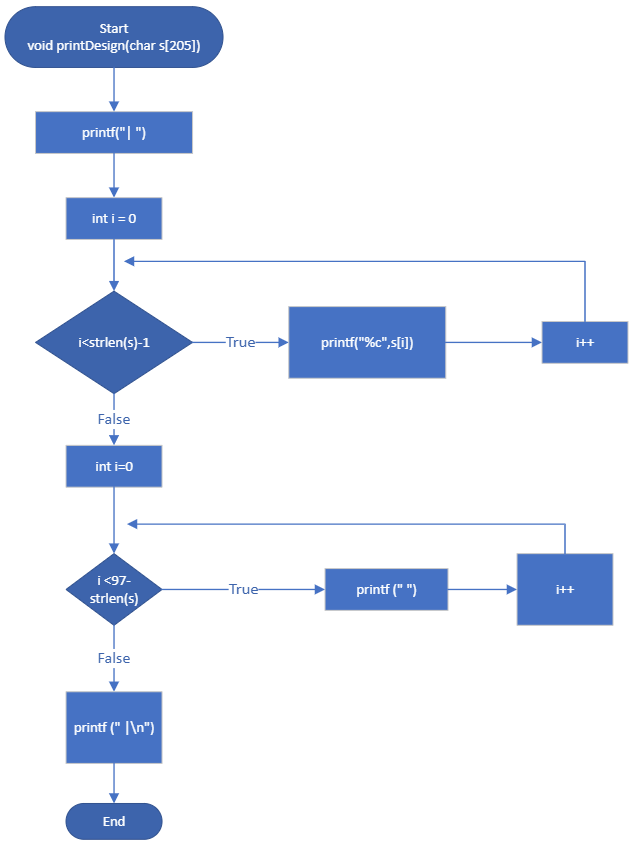
Diagram

Description automatically generated

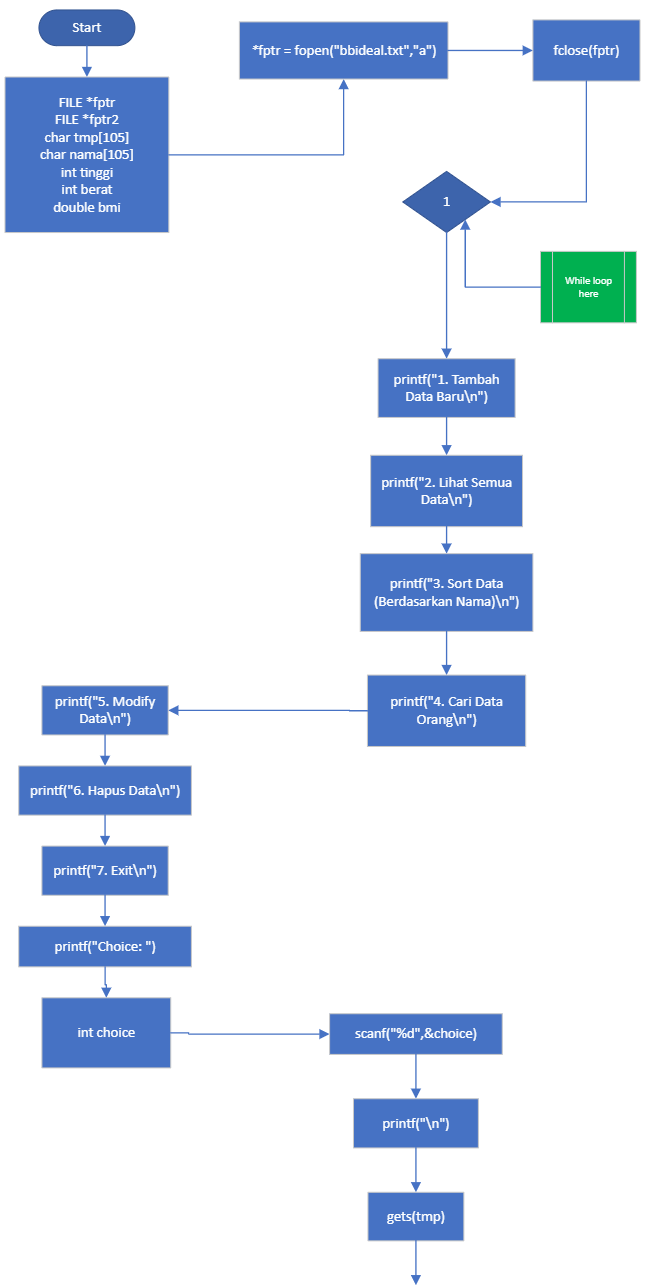
Fungsi ini akan mengcompare string yang jika hasilnya salah akan me return nilai -1, dan jika benar akan mereturn angka dimana jika angka yang di return bukan -1 program yang memakai fungsi ini akan melanjutkan proses.

Fungsi untuk print design data

Fungsi ini akan mengeprint design/border dari tampilan program, yang akan memperbagus tampilan program tersebut



int main()



Fungsi ini berisikan menu yang akan ditampilkan agar user bisa menginput menu mana yang akan dipilih, lalu fungsi akan menuju ke selection

Graphical user interface, diagram, Teams

Description automatically generatedSelection 1 = Tambah data

Jika user memilih 1, maka akan diarahkan ke menu tambah data dimana user diminta untuk menginput data diri berupa nama, berat badan dan tinggi badan, lalu akan di komputasi oleh sistem untuk menghasilkan body mass index dan batas berat badan ideal (min – max), lalu berdasarkan berat badan yang di input user akan dicari tahu apakah user terlalu kurus, normal atau obesitas, setelah semua itu semua data akan di masukkan ke dalam file. Setelah itu fungsi akan berakhir dan user akan diarahkan kembali ke menu utama.

Selection 2 = Lihat data

Jika user memilih 2, maka user akan di arahkan ke menu dimana user dapat melihat data yang telah di input, dan juga hasil perhitungan dari menu selection 1, disini juga akan ditampilkan desain dari fungsi printDesign. Program akan berhenti setelah menampilkan semua data, lalu user akan diarahkan kembali ke menu utama.

Diagram

Description automatically generated

Selection 3 = Sorting berdasarkan nama

A screenshot of a computer screen

Description automatically generated with medium confidence

Jika user memilih 3, akan dijalankan fungsi untuk mengurutkan data data berdasarkan nama dengan bantuan file txt kedua yaitu “bbideal2.txt” untuk menampung data sementara. Setelah data diurutkan akan ditampilkan pesan bahwa data sudah diurutkan, lalu program akan berhenti dan user akan diarahkan kembali ke menu utama.

Selection 4 = Mencari data orang

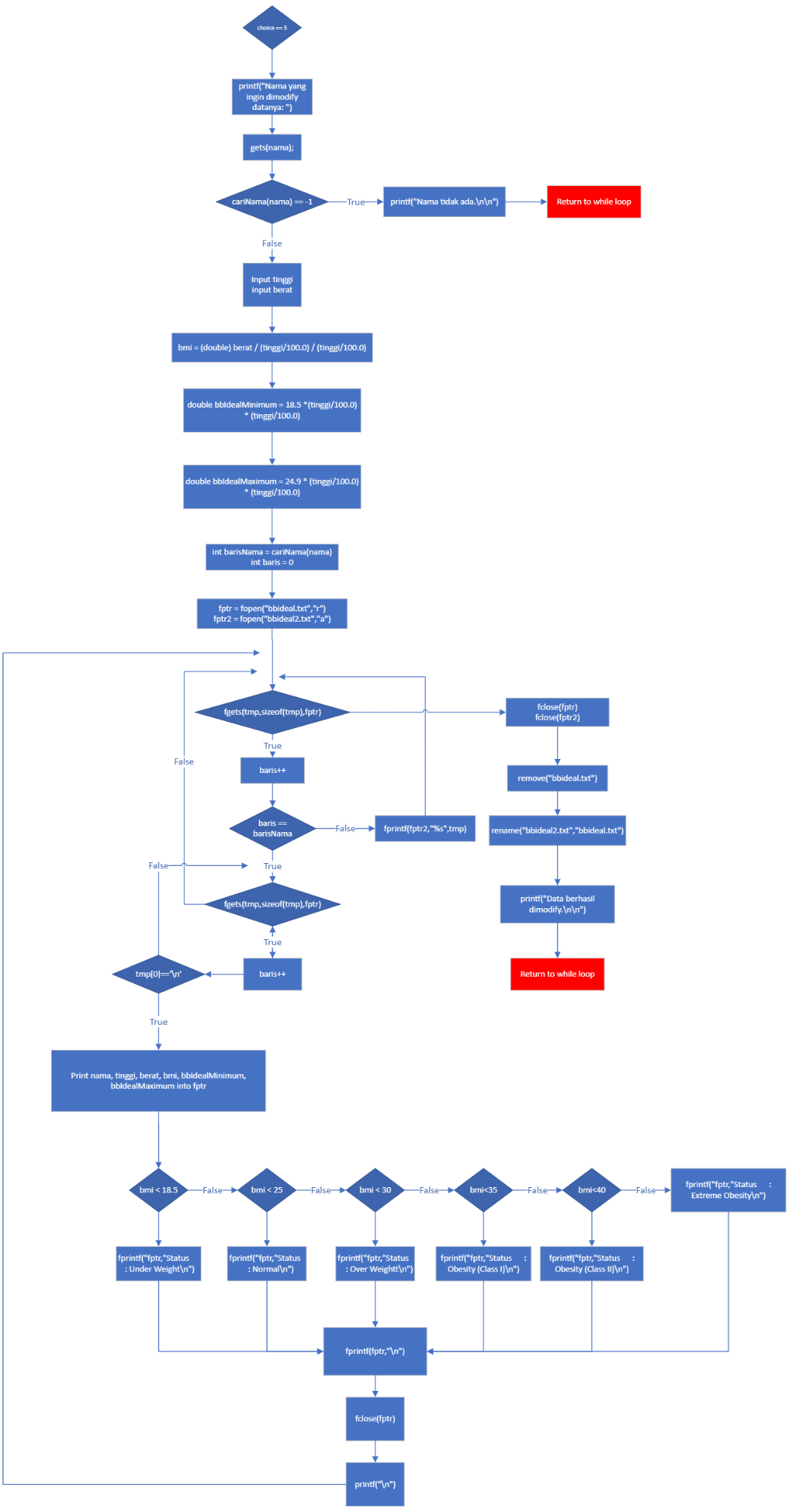
Jika user memilih 4, maka akan diarahkan ke menu dimana user akan menginput nama yang akan dicari, lalu sistem akan memproses input user dan menampilkan hasil output berupa data dari nama yang di input. Jika nama tidak ditemukan, user akan dikembalikan ke menu utama. Kelemahan fungsi ini adalah user diminta untuk menuliskan nama dengan persis, karena jika salah 1 huruf saja data tidak dapat ditampilkan. Program lalu akan berhenti dan user akan dikembalikan ke menu utama

Diagram

Description automatically generated

Selection 5 = Modify data

Jika user memilih 5, maka user akan diarahkan ke menu dimana user akan menginput nama untuk mengubah data dari nama yang di input, dan nama yang di input harus sama persis, karena jika tidak data tidak akan ditemukan, dan jika tidak ada data yang ditemukan user akan dikembalikan ke menu utama. Setelah data ditemukan, user akan diminta untuk menginput data berat badan dan tinggi badan yang baru, lalu akan di proses ulang oleh sistem dengan rumus yang sama dengan rumus di fungsi selection pertama, lalu data akan di write ke dalam file. Fungsi akan berakhir lalu user akan dikembalikan ke menu utama.



Selection 6 = Hapus data

Jika user memilih 6, user akan diarahkan ke menu dimana user akan menginput nama dari data yang ingin dihapus, jika nama tidak ditemukan user akan dikembalikan ke menu utama. Jika nama ditemukan sistem akan melakukan processing yang berujung pada penghapusan data dari nama yang di input user. Fungsi ini akan berhenti setelah selesai menampilkan teks yang berisi bahwa data telah dihapus, lalu user akan dikembalikan ke menu utama.

Diagram

Description automatically generated

Diagram

Description automatically generatedSelection 7 = Exit program

Jika user memilih 7, maka program akan berhenti setelah menampilkan pesan bahwa program telah berakhir.

Bab 3

Program dan Penjelasannya

#include<stdio.h>

#include<stdbool.h>

#include<string.h>

#include<stdlib.h>

//mereturn baris dimana nama yang dicari, apabila tidak ada return -1

int cariNama(char nama[105])

{

char namaFormat[105]="Nama : ";

strcat(namaFormat, nama);

strcat(namaFormat, "\n");

int lineNama = -1;

int line = 0;

char tmp[105];

FILE \*fptr = fopen("bbideal.txt","r");

while(fgets(tmp,sizeof(tmp),fptr))

{

++line;

if(strcmp(namaFormat,tmp) == 0)

{

lineNama = line;

break;

}

}

fclose(fptr);

return lineNama;

}

//untuk print agar rapi

void printDesign(char s[205])

{

printf("| ");

for(int i=0;i<strlen(s)-1;i++) printf("%c",s[i]);

for(int i=0;i<97-strlen(s);i++) printf(" ");

printf(" |\n");

}

int main()

{

FILE \*fptr;

FILE \*fptr2;

char tmp[105];

char nama[105];

int tinggi, berat;

double bmi;

fptr = fopen("bbideal.txt","a");

fclose(fptr);

while(1)

{

//display menu

printf("1. Tambah Data Baru\n");

printf("2. Lihat Semua Data\n");

printf("3. Sort Data (Berdasarkan Nama)\n");

printf("4. Cari Data Orang\n");

printf("5. Modify Data\n");

printf("6. Hapus Data\n");

printf("7. Exit\n");

printf("Choice: ");

int choice;

scanf("%d",&choice);

printf("\n");

gets(tmp);

//menambah data baru

if(choice == 1)

{

system("cls");

printf("Silahkan input data berikut.\n");

printf("Nama: ");

gets(nama);

if(cariNama(nama) != -1)

{

printf("Nama sudah ada di dalam data, silahkan ke menu modify data.\n\n");

system("pause");

system("cls");

continue;

}

printf("Tinggi badan (dalam cm): ");

scanf("%d",&tinggi);

printf("Berat badan (dalam kg): ");

scanf("%d",&berat);

//hitung bmi dan range bb ideal

bmi = (double) berat / (tinggi / 100.0) / (tinggi / 100.0);

double bbIdealMinimum = 18.5 \* (tinggi / 100.0) \* (tinggi / 100.0);

double bbIdealMaximum = 24.9 \* (tinggi / 100.0) \* (tinggi / 100.0);

//tulis ke dalam file

fptr = fopen("bbideal.txt","a");

fprintf(fptr,"Nama : %s\n",nama);

fprintf(fptr,"Tinggi badan : %d cm\n",tinggi);

fprintf(fptr,"Berat badan : %d kg\n",berat);

fprintf(fptr,"BMI : %.2f\n",bmi);

fprintf(fptr,"Berat badan ideal : %.2f - %.2f kg\n",bbIdealMinimum, bbIdealMaximum);

if(bmi < 18.5) fprintf(fptr,"Status : Under Weight\n");

else if(bmi < 25) fprintf(fptr,"Status : Normal\n");

else if(bmi < 30) fprintf(fptr,"Status : Over Weight\n");

else if(bmi < 35) fprintf(fptr,"Status : Obesity (Class I)\n");

else if(bmi < 40) fprintf(fptr,"Status : Obesity (Class II)\n");

else fprintf(fptr,"Status : Extreme Obesity\n");

fprintf(fptr,"\n");

fclose(fptr);

printf("\n");

system("pause");

system("cls");

}

//menampilkan semua data

else if(choice == 2)

{

system("cls");

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

printf("| |\n");

printf("| ~~~~~~~~~~~~~~~~~~~~~~~~~~~~ |\n");

printf("| ~ DATA BERAT BADAN IDEAL ~ |\n");

printf("| ~~~~~~~~~~~~~~~~~~~~~~~~~~~~ |\n");

printf("| |\n");

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

fptr = fopen("bbideal.txt","r");

while(fgets(tmp,sizeof(tmp),fptr))

{

printDesign(tmp);

}

fclose(fptr);

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

system("pause");

system("cls");

}

//sort data berdasarkan nama

else if(choice == 3)

{

system("cls");

char isiBaris[1005][105];

int baris = 0;

//simpan isi file dalam array isiBaris

fptr = fopen("bbideal.txt","r");

while(fgets(tmp,sizeof(tmp),fptr))

{

baris++;

strcpy(isiBaris[baris],tmp);

}

fclose(fptr);

//cari banyak orang dan baris awal dan akhir data nama orang tersebut

int banyakOrang = 0;

int barisMulai[1005];

int barisAkhir[1005];

for(int i=1;i<=baris;i++)

{

if(isiBaris[i][0] == 'N')

{

banyakOrang++;

barisMulai[banyakOrang] = i;

}

}

for(int i=1;i<banyakOrang;i++) barisAkhir[i] = barisMulai[i+1] - 1;

barisAkhir[banyakOrang] = baris;

//proses sorting berdasarkan nama

for(int i=1;i<banyakOrang;i++)

{

for(int j=1;j<banyakOrang;j++)

{

if(strcmp(isiBaris[barisMulai[j]],isiBaris[barisMulai[j+1]]) > 0)

{

int tmpint = barisMulai[j];

barisMulai[j] = barisMulai[j+1];

barisMulai[j+1] = tmpint;

tmpint = barisAkhir[j];

barisAkhir[j] = barisAkhir[j+1];

barisAkhir[j+1] = tmpint;

}

}

}

//isi ke file baru nama yang sudah disort

fptr2 = fopen("bbideal2.txt","a");

for(int i=1;i<=banyakOrang;i++)

{

for(int j=barisMulai[i];j<=barisAkhir[i];j++)

{

fprintf(fptr2,"%s",isiBaris[j]);

}

}

fclose(fptr2);

//file lama direplace dengan file baru

remove("bbideal.txt");

rename("bbideal2.txt","bbideal.txt");

printf("Data berhasil disort berdasarkan nama.\n\n");

system("pause");

system("cls");

}

//cari data orang tertentu

else if(choice == 4)

{

system("cls");

printf("Nama yang ingin dicari: ");

gets(nama);

if(cariNama(nama) == -1)

{

printf("Nama tidak ada.\n\n");

continue;

}

int barisNama = cariNama(nama);

int baris = 0;

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

fptr = fopen("bbideal.txt","r");

while(fgets(tmp,sizeof(tmp),fptr))

{

baris++;

if(baris == barisNama)

{

printDesign(tmp);

while(fgets(tmp,sizeof(tmp),fptr))

{

if(tmp[0] == '\n') break;

printDesign(tmp);

}

break;

}

}

fclose(fptr);

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

system("pause");

system("cls");

}

//modify data orang tertentu

else if(choice == 5)

{

system("cls");

printf("Nama yang ingin dimodify datanya: ");

gets(nama);

if(cariNama(nama) == -1)

{

printf("Nama tidak ada.\n\n");

continue;

}

printf("Tinggi badan (dalam cm): ");

scanf("%d",&tinggi);

printf("Berat badan (dalam kg): ");

scanf("%d",&berat);

//hitung bmi dan range bb ideal

bmi = (double) berat / (tinggi / 100.0) / (tinggi / 100.0);

double bbIdealMinimum = 18.5 \* (tinggi / 100.0) \* (tinggi / 100.0);

double bbIdealMaximum = 24.9 \* (tinggi / 100.0) \* (tinggi / 100.0);

int barisNama = cariNama(nama);

int baris = 0;

fptr = fopen("bbideal.txt","r");

fptr2 = fopen("bbideal2.txt","a");

while(fgets(tmp,sizeof(tmp),fptr))

{

baris++;

if(baris == barisNama)

{

fprintf(fptr2,"%s",tmp);

while(fgets(tmp,sizeof(tmp),fptr))

{

baris++;

//tambahkan data pada tempat nama yang dicari

if(tmp[0] == '\n')

{

fprintf(fptr2,"Tinggi badan : %d cm\n",tinggi);

fprintf(fptr2,"Berat badan : %d kg\n",berat);

fprintf(fptr2,"BMI : %.2f\n",bmi);

fprintf(fptr2,"Berat badan ideal : %.2f - %.2f kg\n",bbIdealMinimum, bbIdealMaximum);

if(bmi < 18.5) fprintf(fptr2,"Status : Under Weight\n");

else if(bmi < 25) fprintf(fptr2,"Status : Normal\n");

else if(bmi < 30) fprintf(fptr2,"Status : Over Weight\n");

else if(bmi < 35) fprintf(fptr2,"Status : Obesity (Class I)\n");

else if(bmi < 40) fprintf(fptr2,"Status : Obesity (Class II)\n");

else fprintf(fptr2,"Status : Extreme Obesity\n");

fprintf(fptr2,"\n");

break;

}

fprintf(fptr2,"%s",tmp);

}

}

else

{

fprintf(fptr2,"%s",tmp);

}

}

fclose(fptr);

fclose(fptr2);

//replace file lama dengan file baru

remove("bbideal.txt");

rename("bbideal2.txt","bbideal.txt");

printf("Data berhasil dimodify.\n\n");

system("pause");

system("cls");

}

//hapus data orang tertentu

else if(choice == 6)

{

system("cls");

printf("Nama yang ingin dihapus: ");

gets(nama);

if(cariNama(nama) == -1)

{

printf("Nama tidak ada.\n\n");

continue;

}

int barisNama = cariNama(nama);

int baris = 0;

fptr = fopen("bbideal.txt","r");

fptr2 = fopen("bbideal2.txt","a");

while(fgets(tmp,sizeof(tmp),fptr))

{

baris++;

if(baris == barisNama)

{

while(fgets(tmp,sizeof(tmp),fptr))

{

baris++;

if(tmp[0]=='\n') break;

}

}

else

{

fprintf(fptr2,"%s",tmp);

}

}

fclose(fptr);

fclose(fptr2);

//replace file lama dengan file baru

remove("bbideal.txt");

rename("bbideal2.txt","bbideal.txt");

printf("Data berhasil terhapus.\n\n");

system("pause");

system("cls");

}

//exit program

else if(choice == 7)

{

printf("Program berakhir.\n");

break;

}

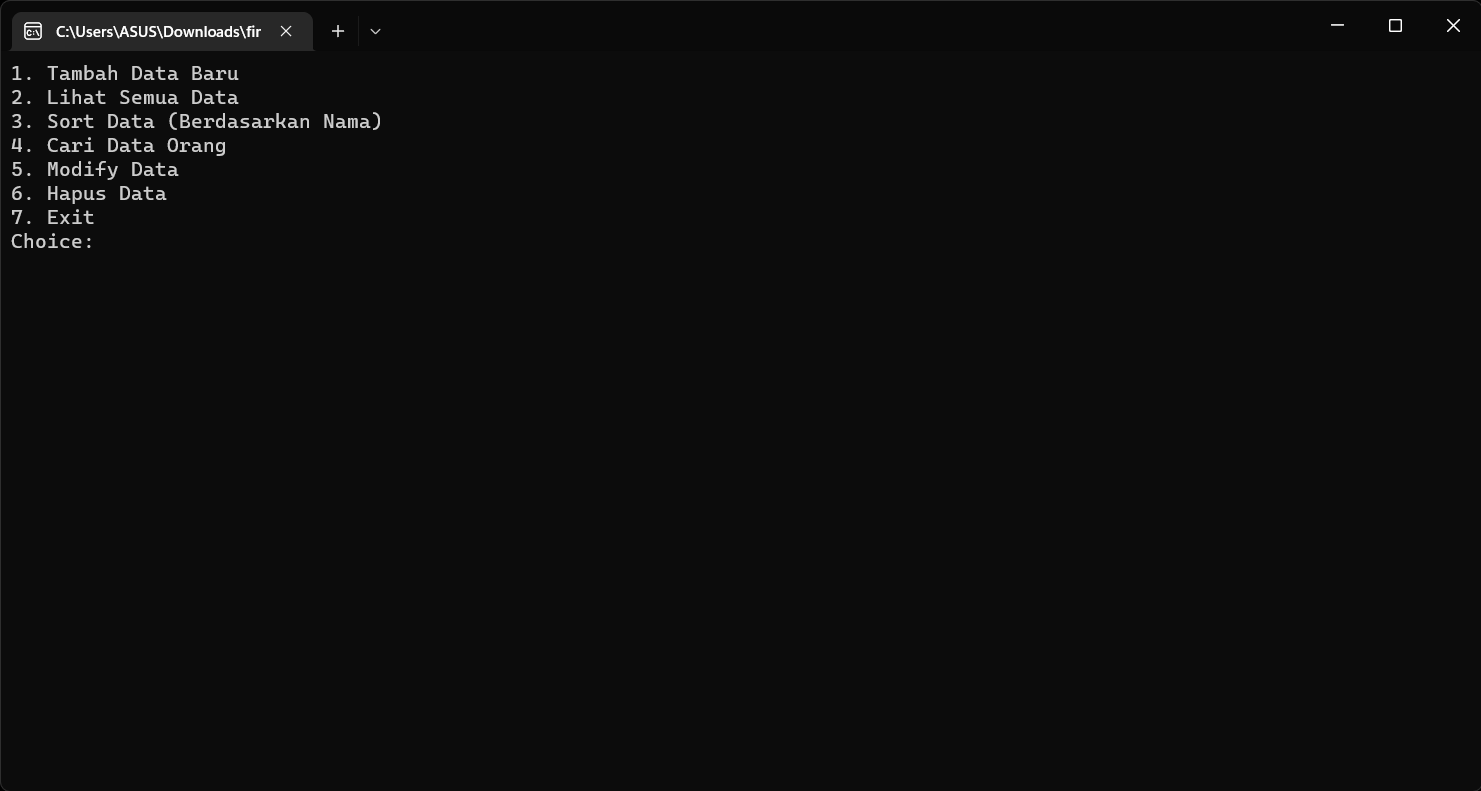
}

}

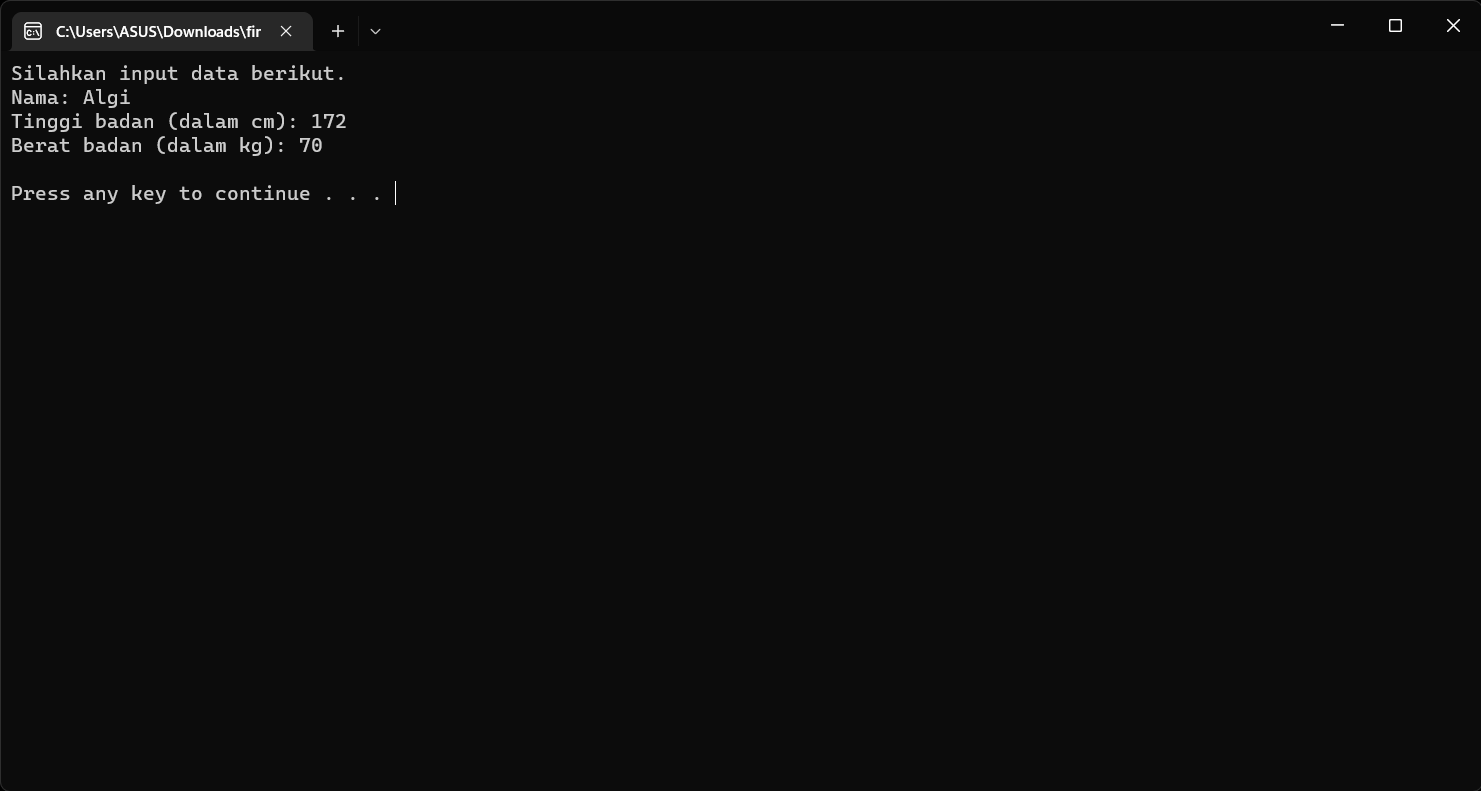
Bab 4

Tampilan Hasil

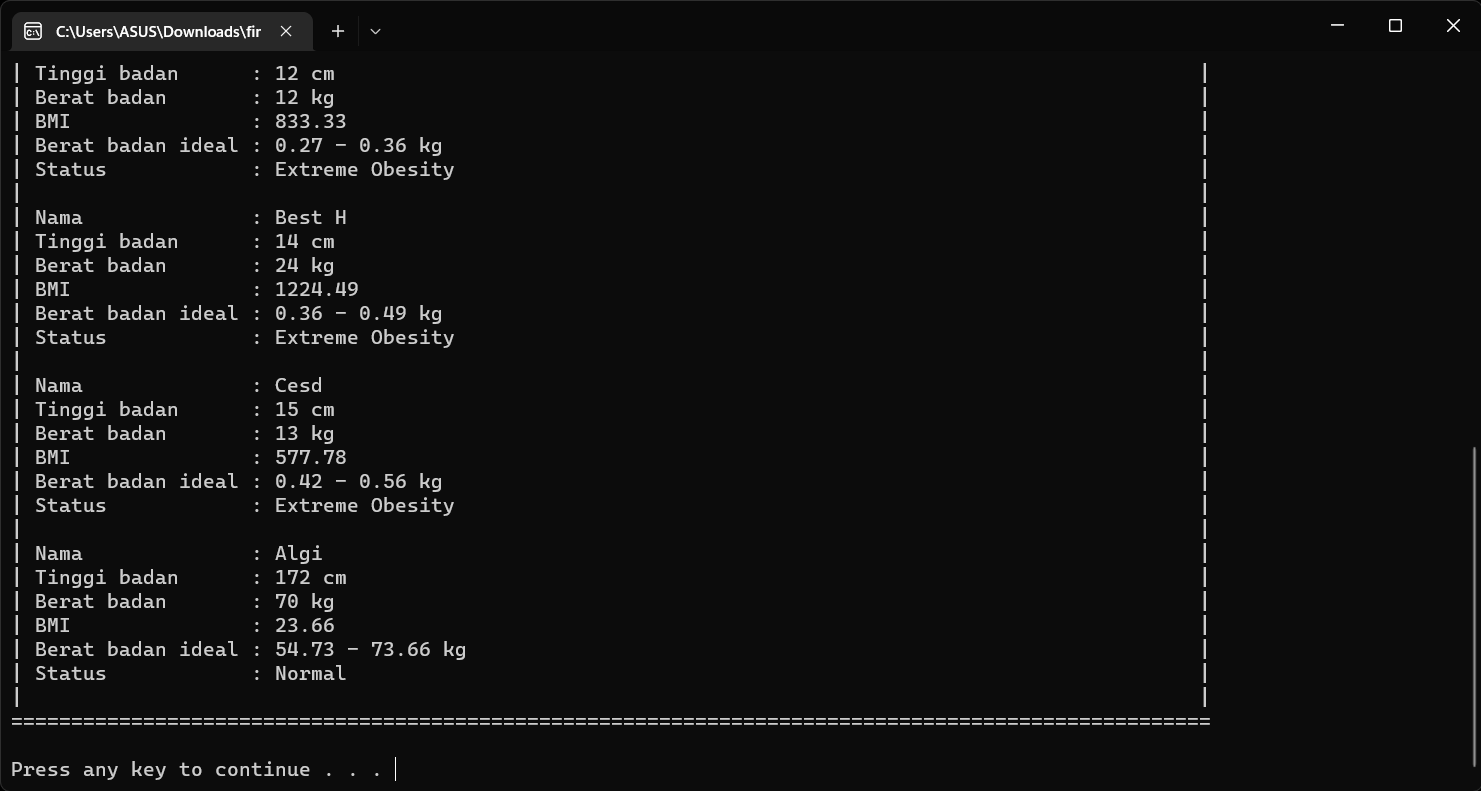
Main menu



Menu Tambah Data Baru



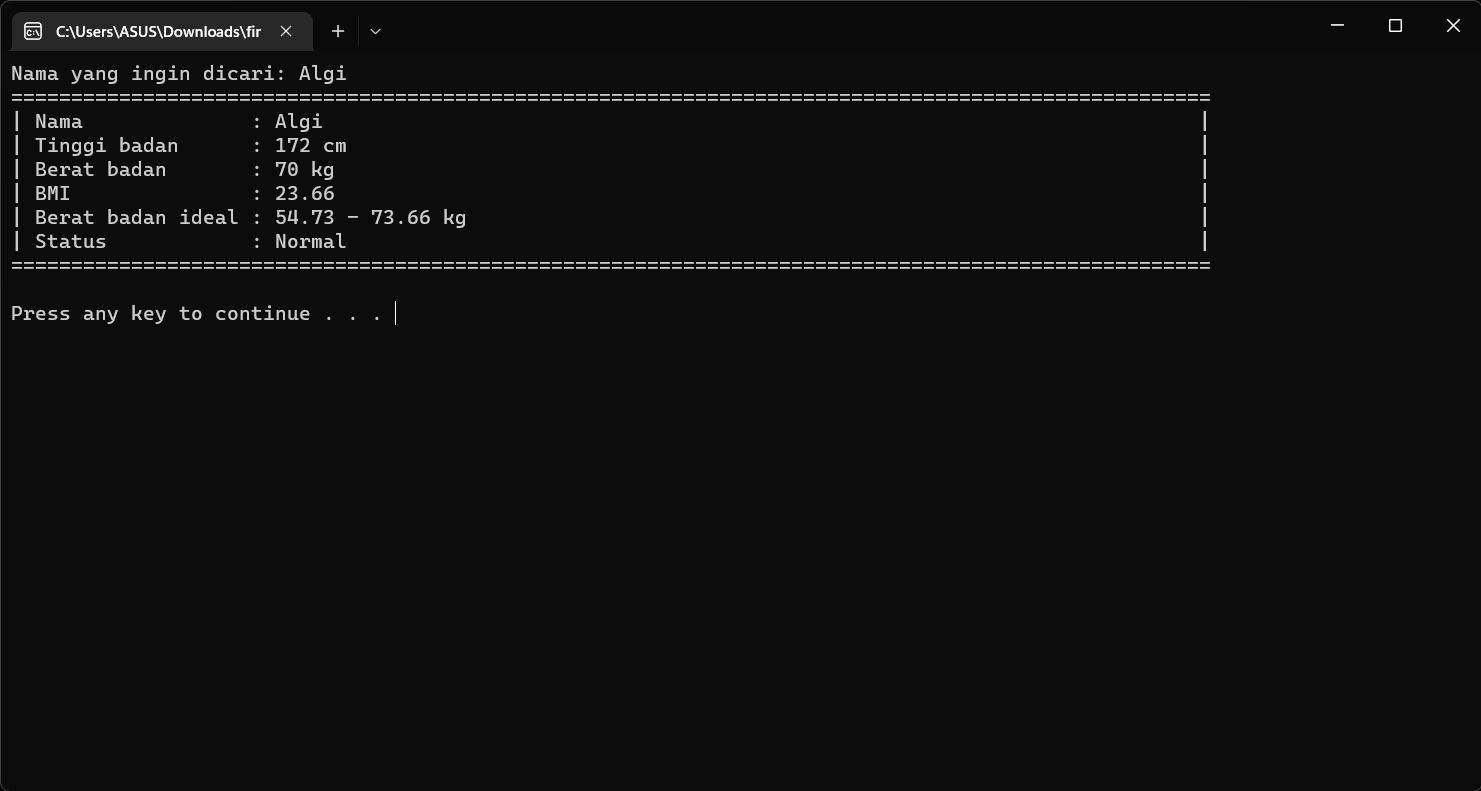
Menu Lihat Semua Data

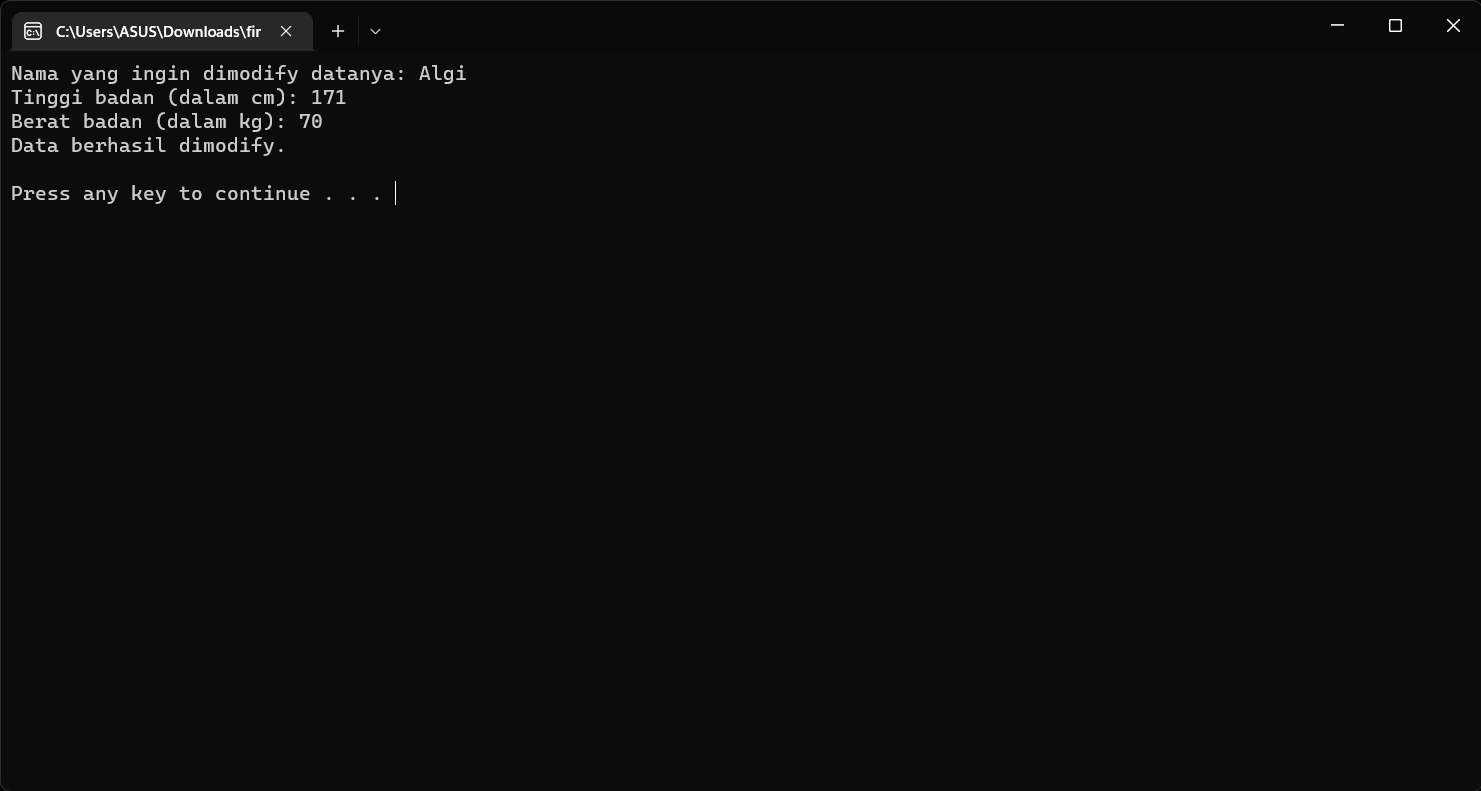


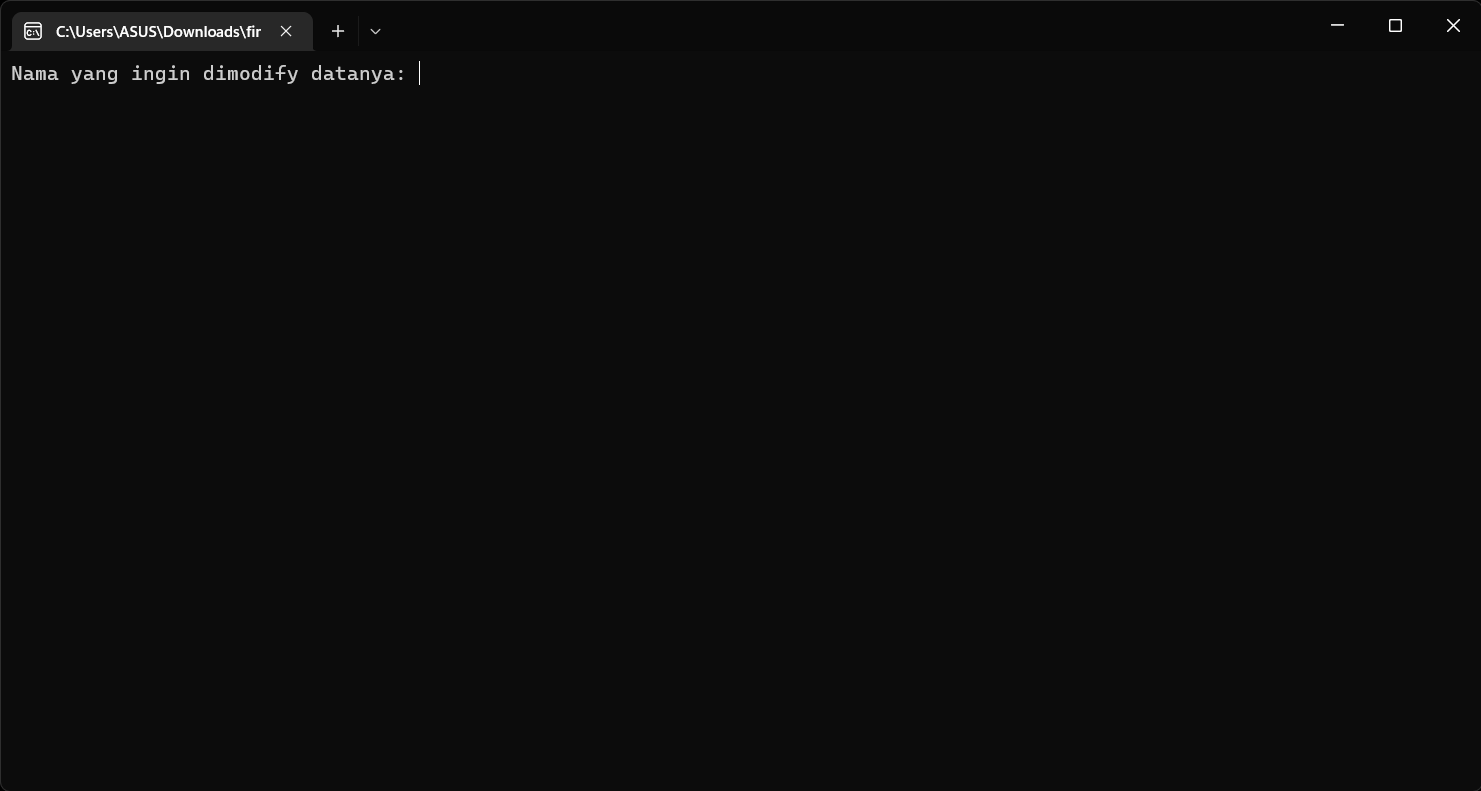
Menu Sorting Data Berdasarkan Nama



Menu Cari Data Orang



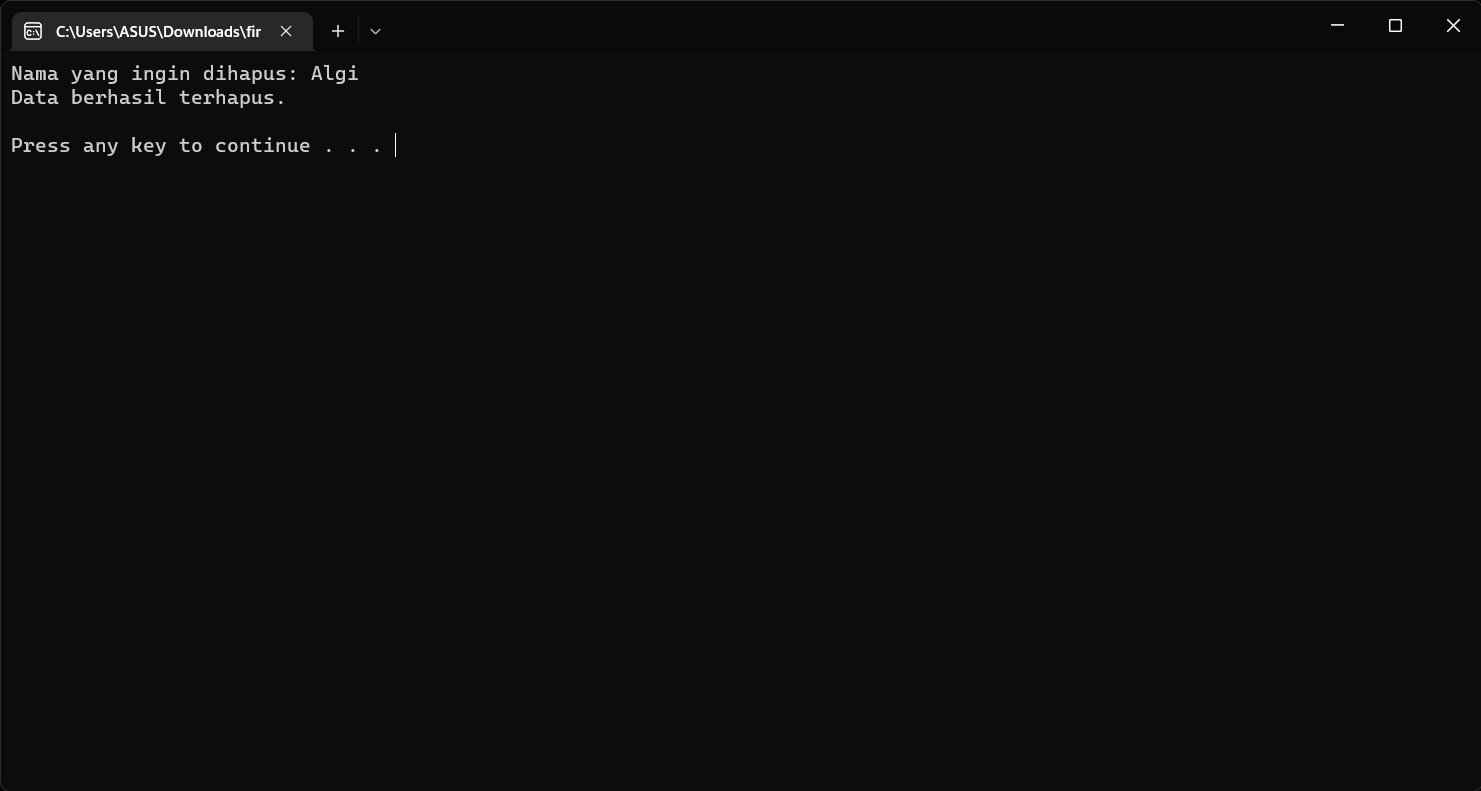
Menu Modify Data



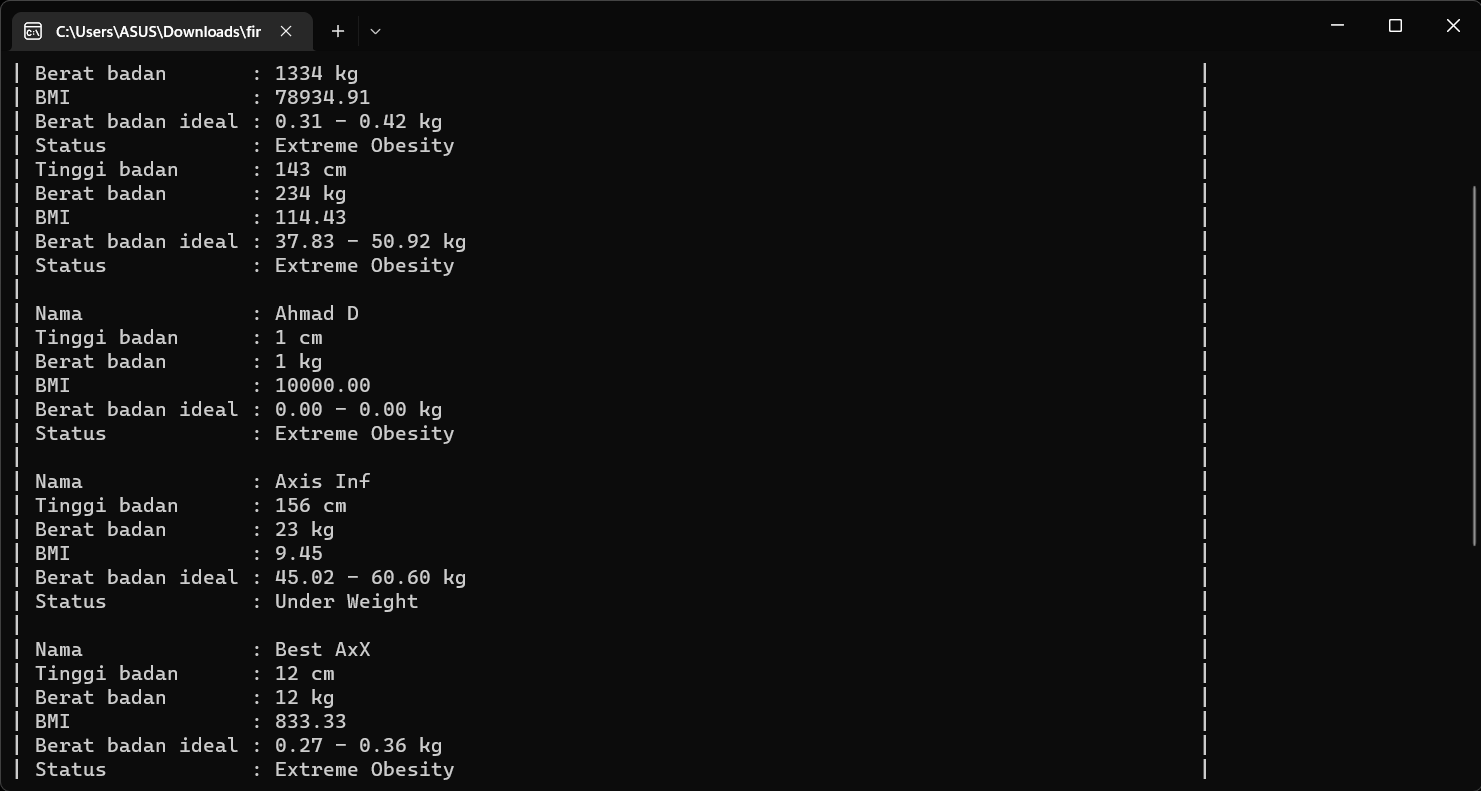
Menu Lihat Semua Data (setelah data di modify)

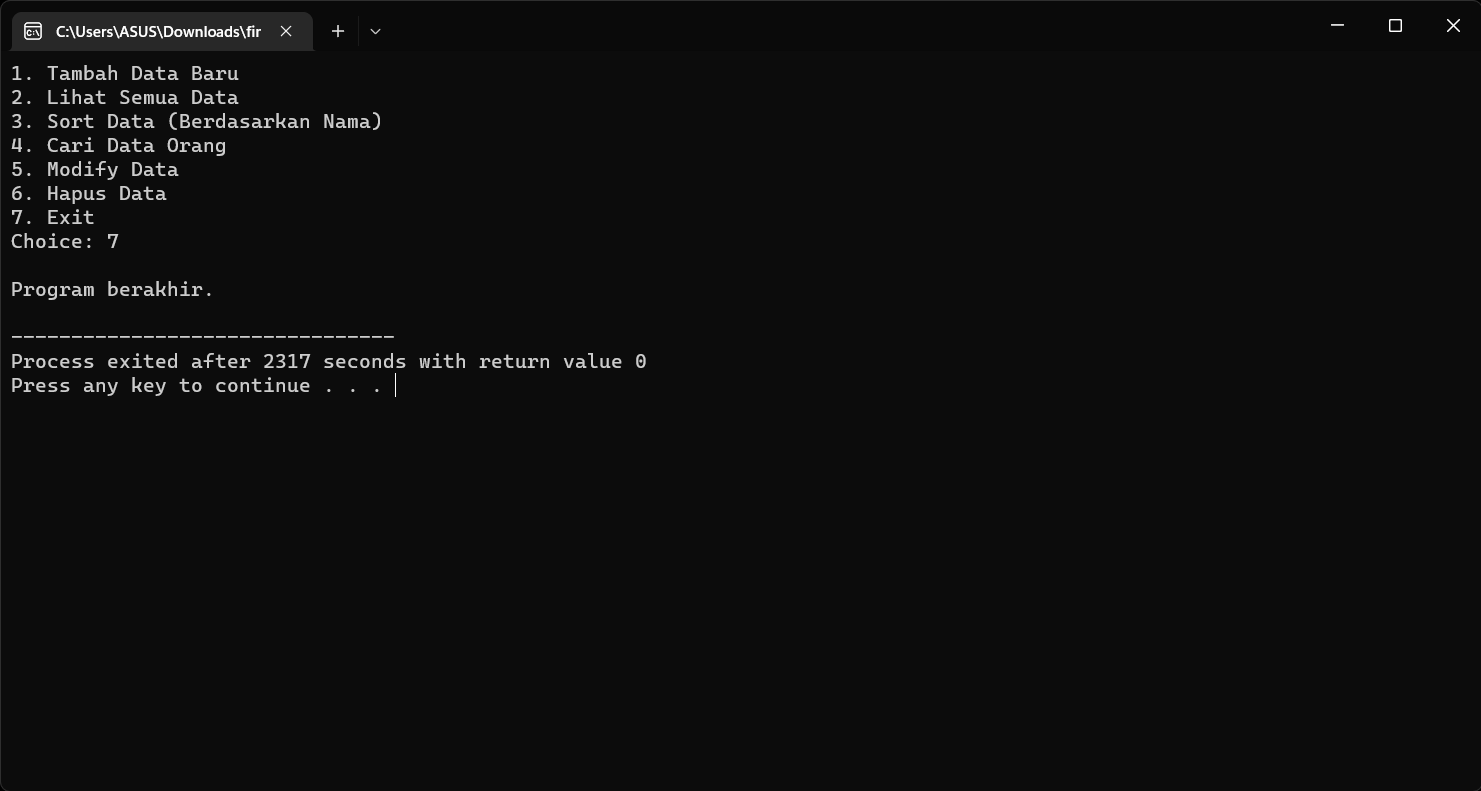


Menu Hapus Data



Menu Lihat Semua Data (setelah data dihapus)



Menu Exit

Bab 5

Kesimpulan

IBWCalc adalah program yang mampu menunjukkan seberapa ideal tubuh kita berdasarkan tinggi dan berat badan yang kita input ke dalam program ini. Fitur fitur pemrograman dalam bahasa C seperti file processing dan struct digunakan untuk membantu pembuatan program Ideal Body Weight Calculator ini. Kedepannya fitur program ini dapat ditambahkan agar dalam penggunaan nya dapat dipakai dengan lebih efisien dan fleksibel.