Laporan Makassar 17/06/2020

**Laporan Tugas Proyek**

**“Menghitung Nilai Akhir Mahasiswa”**



MUH. FACHRISYAM

13020190149

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS MUSLIM INDONESIA**

**MAKASSAR**

**2020**

**PENDAHULUAN**

**A. Deskripsi Kasus**

a. Pengguna memasukkan larik/array jumlah mahasiswa kemudian mengisi nama,stambuk beserta komponen nilai masing-masing mahasiswa

b. Komponen Penilaian: Nilai UTS, UAS, Tugas Proyek dan Tugas Mandiri/Kuis dan Kehadiran

c. Menghitung Komponen Nilai (Operator dan Ekspresi)  
o Nilai UTS = Nilai UTS dikali dengan 20%  
o Nilai UAS = Nilai UAS dikali dengan 25%  
o Nilai Tugas Mandiri = Nilai Tugas Mandiri dikali dengan 20%  
o Nilai Tugas Proyek = Nilai Tugas Proyek dikali dengan 20%  
o Kehadiran = Nilai Kehadiran dikali dengan 15%

d. Menghitung Nilai Akhir = Tugas Mandiri/kuis+Tugas Proyek+UTS+UAS+Kehadiran

e. Konversi Nilai Akhir ke dalam Huruf :

Final Score Nilai Akhir Nilai Konversi > 85 (\*) memperoleh nilai A (4.00 )

                                                     81 - 85 memperoleh nilai A- (3.75 )

                                                     76 -80 memperoleh nilai B+ (3.50)

                                                     71-  75 memperoleh nilai B (3.00)

                                                     66 -70 memperoleh nilai B- (2.75)

                                                     61 - 65 memperoleh nilai C+ (2.50 )

                                                     51 - 60 memperoleh nilai C (2.00 )

                                                    45 - 50 memperoleh nilai D (1.00)

                                                        < 45 memperoleh nilai E (0.00)

ket: (\*)mahasiswa kehadirannya >90%, dan tidak ada komponen nilai bernilai Nol

f. Menampilkan stambuk,nilai akhir,ipk,predikat mahasiswa

g. Mengurutkan data berdasarkan nilai akhir untuk menentukan nilai tertinggi

h. Mencari data berdasarkan nama mahasiswa

i. Menampilkan pembagian nilai semua mahasiswa

**B. Algoritma**

* Mulai
* Memasukkan jumlah data mahasiswa
* Menginput data beserta komponen nilai setiap mahasiswa sebanyak jumlah data
* Memanggil fungsi hitungNilaiAkhir
  + Nilai UTS = Nilai UTS dikali dengan 20%
  + Nilai UAS = Nilai UAS dikali dengan 25%
  + Nilai Tugas Mandiri = Nilai Tugas Mandiri dikali dengan 20%
  + Nilai Tugas Proyek = Nilai Tugas Proyek dikali dengan 20%
  + Kehadiran = Nilai Kehadiran dikali dengan 15%
  + Menghitung Nilai Akhir = Tugas Mandiri/kuis+Tugas Proyek+UTS+UAS+Kehadiran
  + Jika, Mahasiswa memiliki nilaiKehadiran kurang dari 90% dari total kehadiran atau salah satu dari komponen penilaian bernilai Nol, Maka akan mendapat IPK 0 dengan predikat E
  + Jika, Mahasiswa memiliki nilaiAkhir diatas dari 85, Maka akan mendapat IPK 4 dengan predikat A
  + Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 81 dan lebih kecil atau sama dengan 85, Maka akan mendapat IPK 3,75 dengan predikat A-
  + Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 76 dan lebih kecil atau sama dengan 80, Maka akan mendapat IPK 3,50 dengan predikat B+
  + Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 71 dan lebih kecil atau sama dengan 75, Maka akan mendapat IPK 3 dengan predikat B
  + Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 66 dan lebih kecil atau sama dengan 70, Maka akan mendapat IPK 2,75 dengan predikat B-
  + Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 61 dan lebih kecil atau sama dengan 65, Maka akan mendapat IPK 2,50 dengan predikat C+
  + Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 51 dan lebih kecil atau sama dengan 60, Maka akan mendapat IPK 2 dengan predikat C
  + Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 45 dan lebih kecil atau sama dengan 50, Maka akan mendapat IPK 1 dengan predikat D
  + Jika, Mahasiswa memiliki nilaiAkhir kurang dari 45, Maka akan mendapat IPK 0 dengan predikat E
* Memanggil fungsi cetakData
  + Menampilkan nama,stambuk,nilai akhir,ipk,predikat mahasiswa dalam bentuk baris
* Switch Case
  + 1. Memanggil fungsi sorting
  + 2. Memanggil fungsi search
  + 3. Menampilkan pembagian nilai setiap mahasiswa
* Selesai

**C. Implementasi**

#include<iostream>

#include<iomanip>

#include<string>

using namespace std;

struct DataMhs{

string nama;

string stambuk;

float nilaiUTS;

float nilaiUAS;

float nilaiTugasMandiri;

float nilaiTugasProyek;

float nilaiKehadiran;

float nilaiAkhir;

string predikat;

float ipk;

};

void profil()

{

system("cls");

cout<<"============================="<<endl;

cout<<"=== Nama: Muh. Fachrisyam ==="<<endl;

cout<<"=== Stambuk: 13020190149 ==="<<endl;

cout<<"=== Kelas: A3 ==="<<endl;

cout<<"============================="<<endl;

}

void cetakData(DataMhs data[], int &banyak);

void cetakHeadData();

void cetakPembagianNilai(DataMhs data[],int &banyak);

void hitungNilaiAkhir(DataMhs data[], int &banyak);

void sorting(DataMhs data[],int &banyak);

void search(DataMhs data[],DataMhs hasil[], int &banyak, char &pilih);

int main(){

char pilih;

int n;

profil();

cout<<"-=PROGRAM MENGHITUNG NILAI MAHASISWA=-"<<endl<<endl;

cout<<"Masukkan Jumlah Data: ";

cin>>n;

DataMhs mhs[n];

DataMhs pencarian[n];

for(int a=0; a<n ; a++){

cout<<"Data Mahasiswa "<<a+1<<": "<<endl;

cout<<"Stambuk: " ; cin>>mhs[a].stambuk ;

cout<<"Nama: " ;cin.ignore(); getline(cin,mhs[a].nama);

cout<<"Nilai UTS: " ; cin>>mhs[a].nilaiUTS ;

cout<<"Nilai UAS: " ; cin>>mhs[a].nilaiUAS ;

cout<<"Nilai Tugas Mandiri: " ; cin>>mhs[a].nilaiTugasMandiri ;

cout<<"Nilai Tugas Proyek: " ; cin>>mhs[a].nilaiTugasProyek ;

cout<<"Nilai Kehadiran(\*min 90): " ; cin>>mhs[a].nilaiKehadiran ; cout<<endl;

}

hitungNilaiAkhir(mhs,n);

MENU:

cetakHeadData();

cetakData(mhs,n);

cout<<endl;

cout<<"1.Urut berdasarkan nilai akhir(DESC)"<<endl;

cout<<"2.Cari berdasarkan nama"<<endl;

cout<<"3.Cetak pembagian nilai semua Mahasiswa"<<endl;

cout<<"Pilihan: ";

cin>>pilih;

switch (pilih){

case '1':

sorting(mhs,n);

cetakHeadData();

cetakData(mhs,n);

cout<<"\nKembali? (y/n): ";

cin>>pilih;

if(pilih=='y' || pilih=='Y')

goto MENU;

break;

case '2':

search(mhs,pencarian,n,pilih);

cout<<"\nKembali? (y/n): ";

cin>>pilih;

if(pilih=='y' || pilih=='Y')

goto MENU;

break;

case '3':

cetakPembagianNilai(mhs,n);

cout<<"\nKembali? (y/n): ";

cin>>pilih;

if(pilih=='y' || pilih=='Y')

goto MENU;

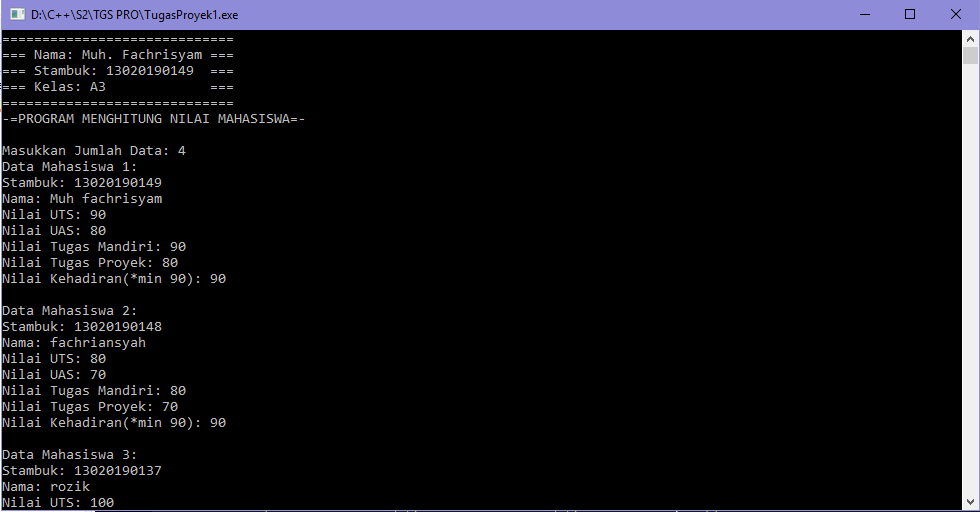
break;

}

return 0;

}

Header #include<iostream> digunakan sebagai standar input dan output ke console, fungsi yang saya gunakan yaitu cout, cin, endl. Header #include<iomanip> digunakan untuk mempermudah dalam menampilkan data kelayar dengan mengatur spasinya menggunakan fungsi setw. Header #include<string> saya gunakan dalam fungsi searching yang dimana menggunakan fungsi find untuk mencari string. Using namespace std agar bisa menggunakan standard library c++. Struct DataMhs berisi komponen-komponen data mahasiswa.



Pertama kita di suruh untuk menginput berapa jumlah data mahasiswanya kedalam variabel n yang nantinya menjadi jumlah array. Kemudian program berjalan melakukan perulangan sebanyak n sesuai inputan, didalam perulangan kita mengisi komponen struct yaitu stambuk,nama,nilai uts, nilai uas,nilai tugas mandiri, nilai tugas proyek, dan nilai kehadiran setiap mahasiswa. Setelah menginput nilai-nilai mahasiswa, program masuk ke fungsi “hitungNilaiAkhir(mhs,n)” dengan mengirim data semua mahasiswa dan banyak mahasiswa sebagai parameter.

void hitungNilaiAkhir(DataMhs data[], int &banyak){

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

data[i].nilaiUTS = data[i].nilaiUTS \* 20/100;

data[i].nilaiUAS = data[i].nilaiUAS \* 25/100;

data[i].nilaiTugasMandiri = data[i].nilaiTugasMandiri \* 20/100;

data[i].nilaiTugasProyek = data[i].nilaiTugasProyek \* 20/100;

data[i].nilaiKehadiran = data[i].nilaiKehadiran \* 15/100;

data[i].nilaiAkhir = data[i].nilaiUTS + data[i].nilaiUAS + data[i].nilaiTugasMandiri

+ data[i].nilaiTugasProyek + data[i].nilaiKehadiran;

if(data[i].nilaiKehadiran < 13.5 || data[i].nilaiUTS == 0 || data[i].nilaiUAS == 0

|| data[i].nilaiTugasMandiri == 0 || data[i].nilaiTugasProyek == 0){

data[i].predikat = "E";

data[i].ipk = 0;

} else if(data[i].nilaiAkhir > 85){

data[i].predikat = "A";

data[i].ipk = 4.00;

} else if(data[i].nilaiAkhir >= 81 && data[i].nilaiAkhir<=85){

data[i].predikat = "A-";

data[i].ipk = 3.75;

} else if(data[i].nilaiAkhir >= 76 && data[i].nilaiAkhir<=80.9){

data[i].predikat = "B+";

data[i].ipk = 3.50;

} else if(data[i].nilaiAkhir >= 71 && data[i].nilaiAkhir<=75.9){

data[i].predikat = "B";

data[i].ipk = 3.00;

} else if(data[i].nilaiAkhir >= 66 && data[i].nilaiAkhir<=70.9){

data[i].predikat = "B-";

data[i].ipk = 2.75;

} else if(data[i].nilaiAkhir >= 61 && data[i].nilaiAkhir<=65.9){

data[i].predikat = "C+";

data[i].ipk = 2.50;

} else if(data[i].nilaiAkhir >= 51 && data[i].nilaiAkhir<=60.9){

data[i].predikat = "C";

data[i].ipk = 2.00;

} else if(data[i].nilaiAkhir >= 45 && data[i].nilaiAkhir<=50.9){

data[i].predikat = "D";

data[i].ipk = 1.00;

} else if(data[i].nilaiAkhir <45){

data[i].predikat = "E";

data[i].ipk = 0;

}

}

}

Dalam fungsi hitungNilaiAkhir , dilakukan perhitungan untuk setiap data mahasiswa sebagai berikut :

Menghitung Komponen Nilai  
o Nilai UTS = Nilai UTS dikali dengan 20%  
o Nilai UAS = Nilai UAS dikali dengan 25%  
o Nilai Tugas Mandiri = Nilai Tugas Mandiri dikali dengan 20%  
o Nilai Tugas Proyek = Nilai Tugas Proyek dikali dengan 20%  
o Kehadiran = Nilai Kehadiran dikali dengan 15%

o Menghitung Nilai Akhir = Tugas Mandiri/kuis+Tugas Proyek+UTS+UAS+Kehadiran

setelah melakukan perhitungan untuk mengisi nilaiAkhir , maka akan memasuki percabangan if else if untuk menentukan IPK dan predikat yang diperoleh mahasiswa dengan ketentuan sebagai berikut :

* Jika, Mahasiswa memiliki nilaiKehadiran kurang dari 90% dari total kehadiran atau salah satu dari komponen penilaian bernilai Nol, Maka akan mendapat IPK 0 dengan predikat E
* Jika, Mahasiswa memiliki nilaiAkhir diatas dari 85, Maka akan mendapat IPK 4 dengan predikat A
* Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 81 dan lebih kecil atau sama dengan 85, Maka akan mendapat IPK 3,75 dengan predikat A-
* Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 76 dan lebih kecil atau sama dengan 80, Maka akan mendapat IPK 3,50 dengan predikat B+
* Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 71 dan lebih kecil atau sama dengan 75, Maka akan mendapat IPK 3 dengan predikat B
* Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 66 dan lebih kecil atau sama dengan 70, Maka akan mendapat IPK 2,75 dengan predikat B-
* Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 61 dan lebih kecil atau sama dengan 65, Maka akan mendapat IPK 2,50 dengan predikat C+
* Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 51 dan lebih kecil atau sama dengan 60, Maka akan mendapat IPK 2 dengan predikat C
* Jika, Mahasiswa memiliki nilaiAkhir lebih besar atau sama dengan 45 dan lebih kecil atau sama dengan 50, Maka akan mendapat IPK 1 dengan predikat D
* Jika, Mahasiswa memiliki nilaiAkhir kurang dari 45, Maka akan mendapat IPK 0 dengan predikat E

Akhir dari fungsi hitungNilaiAkhir.

Program kembali ke main (fungsi utama), lalu menjalankan fungsi “cetakHeadData” dan “cetakData(mhs,n)” mengirim data mahasiswa yang telah dihitung dan banyak array sebagai parameter. Fungsi ini bertujuan untuk menampilkan stambuk, nama, nilai akhir, IPK, dan predikat setiap mahasiswa dalam bentuk baris ke console

void cetakHeadData(){

profil();

cout<<"-------------------------------------------------------------------------------------------------------------"<<endl;

cout<<"No\tStambuk\t\tNama\t\tNilai Akhir IPK Predikat"<<endl;

cout<<"-------------------------------------------------------------------------------------------------------------"<<endl;

}

void cetakData(DataMhs data[], int &banyak){

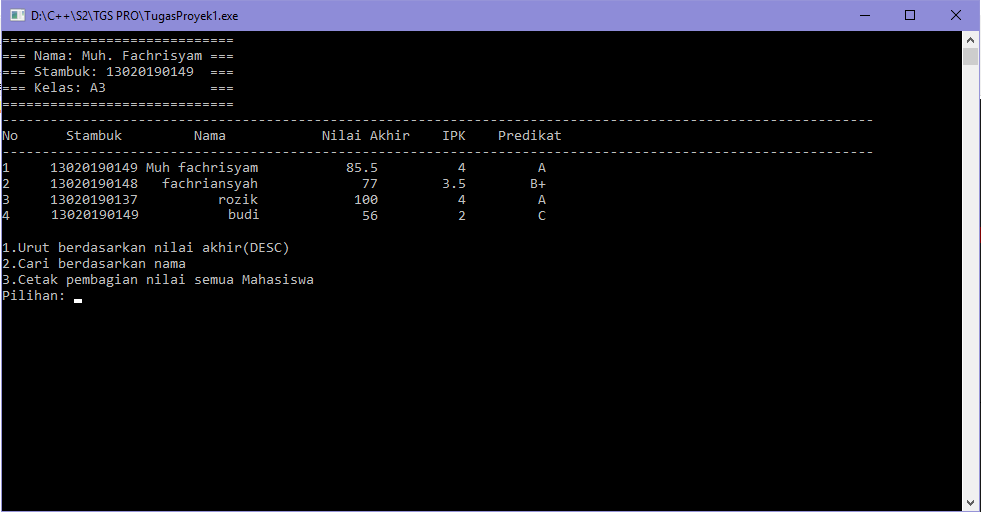
for(int a=0 ; a<banyak ; a++){

cout<<a+1<<setw(16)<<data[a].stambuk<<setw(15)<<data[a].nama<<setw(15)<<data[a].nilaiAkhir

<<setw(11)<<data[a].ipk<<setw(10)<<data[a].predikat<<endl;

}

}



Setelah data dicetak, dalam fungsi main didalam “MENU:” diberikan 3 pilihan; (1)untuk mengurutkan data berdasarkan nilai akhir. Sehingga bisa terlihat jelas siapa yang memiliki nilai yang lebih tinggi(DESCENDING), (2)pilihan untuk mencari data berdasarkan nama yang diinput, (3)pilihan untuk menampilkan pembagian nilai semua mahasiswa. Pemilihan dilakukan oleh percabangan switch case dengan menginput angka 1,2 dan 3.

* Dengan menginput angka 1 maka program akan menjalankan fungsi “sorting(mhs,n)” yang berfungsi mengurutkan nilai akhir secara descending

void sorting(DataMhs data[],int &banyak){

//insertion sort

int i,j;

DataMhs temp;

for(i=1;i<=banyak-1;i++)

{

temp=data[i];

j=i-1; //var j sebagai indeks sebelumnya

while((temp.nilaiAkhir>data[j].nilaiAkhir)&&(j>=0)) //kondisi data ke i kurang dengan data dibelakangnya

{

data[j+1]=data[j]; //Menyimpan data ke i ke indeks sebelumnya

j=j-1; //assign j batas loop

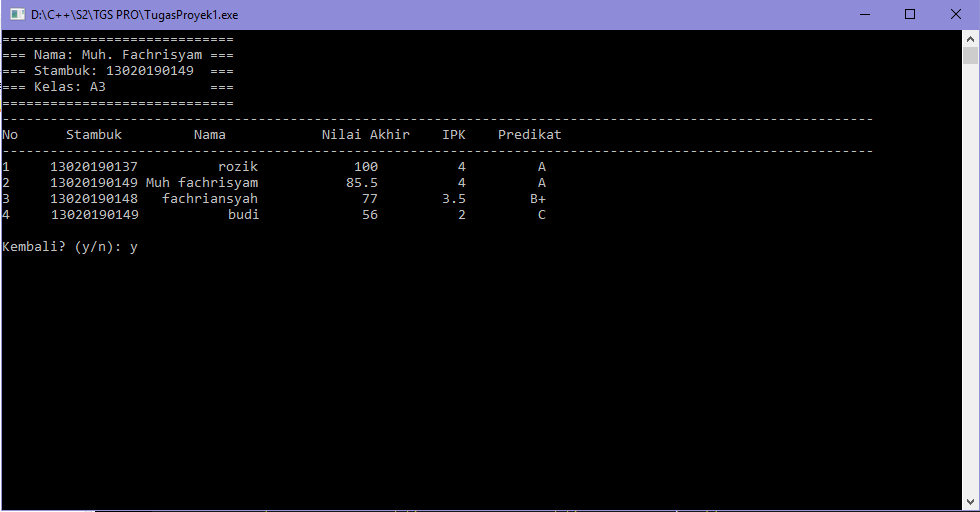
}

data[j+1]=temp; //menyimpan data yang sudah dicek

}

}

Sorting dilakukan dengan cara insertion sort tanpa alasan khusus. Setelah sorting program kembali ke main dan kembali memanggil fungsi “cetakHeadData()” dan “cetakData(mhs,n)” untuk menampilkan data, sebelum keluar dari case 1 diberikan pilihan untuk kembali atau tidak dengan memencet “y/n” pada keyboard. Jika menekan ‘y’ maka program akan kembali ke “MENU:” , jika tidak maka program akan selesai



* Dengan menginput angka 2 , program akan memasuki fungsi “search(mhs,pencarian,n,pilih)” DataMhs mhs ,DataMhs pencarian, n sebagai banyak array , dan char pilih sebagai parameter

void search(DataMhs data[],DataMhs hasil[], int &banyak,char &pilih

int a=0;

size\_t found;

string cari;

cout<<"Masukkan Nama: ";

cin.ignore(); getline(cin,cari);

cetakHeadData();

for(int i=0;i<banyak;i++)

{

found = data[i].nama.find(cari);

if(found != string::npos)

{

hasil[a]=data[i];

a++;

}

}

cetakData(hasil,a);

cout<<"\nPencarian '"<<cari<<"' menemukan "<<a<<" data"<<endl;

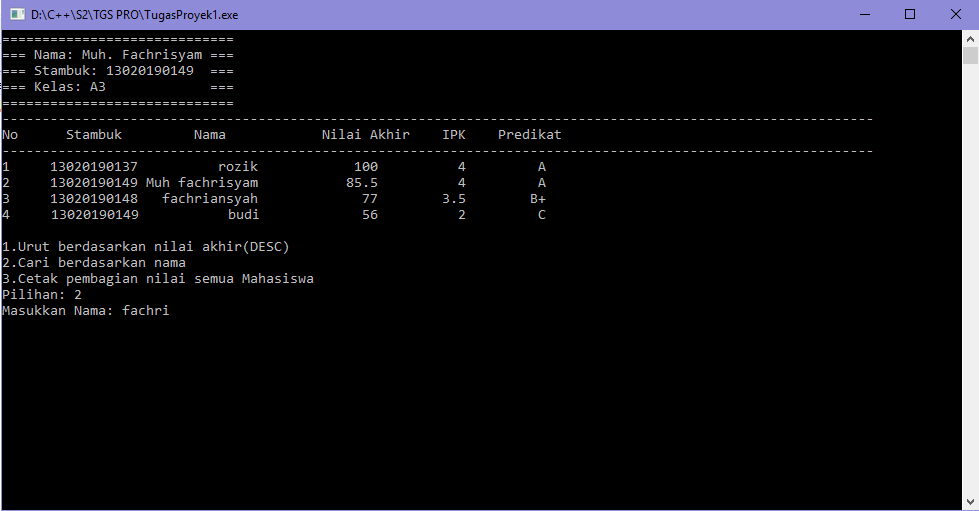
cout<<"Tampilkan pembagian nilai? (y/n): ";

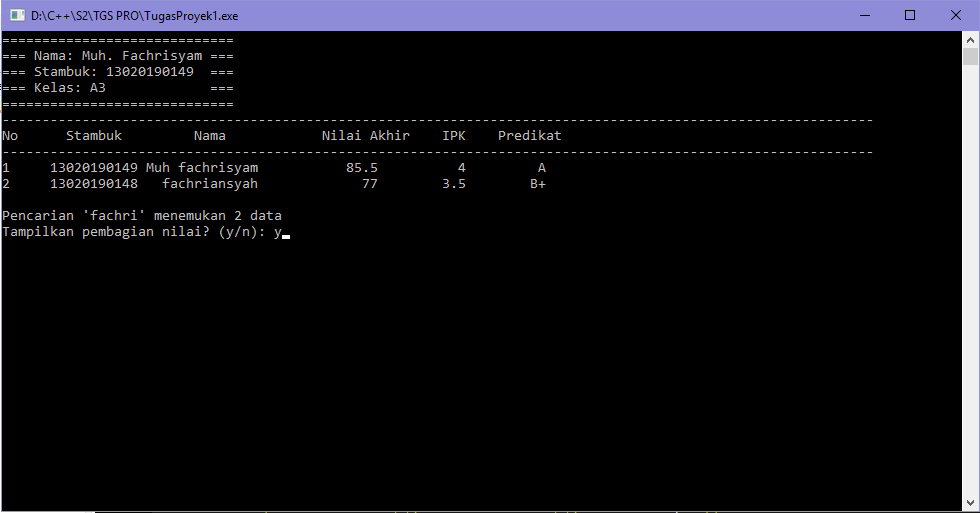
cin>>pilih;

if(pilih=='y' || pilih=='Y')

cetakPembagianNilai(hasil,a);

}





Dalam fungsi ini kita menginput “string cari” yang digunakan untuk mencari nama setiap mahasiswa dalam array, dengan menggunakan fungsi find yang terdapat dalam header string berfungsi untuk menemukan posisi kata yang diinput pada setiap nama mahasiswa dengan perulangan, didalam perulangan terdapat pengecekan dengan if.

* + jika ditemukan nama yang mengandung kata yang telah diinput maka akan menyimpan datanya kedalam DataMhs hasil

setelah itu akan ada pilihan untuk menampilkan pembagian nilai mahasiswa dari hasil pencarian tersebut dengan menginput (y/n) pada char pilih, jika ‘y’ maka akan memanggil fungsi “cetakPembagianNilai(hasil,a)” untuk mencetak semua komponen dari struct DataMhs hasil.

void cetakPembagianNilai(DataMhs data[],int &banyak){

profil();

cout<<"\n-=Pembagian Nilai=-"<<endl;

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

cout<<"\n-Mahasiswa "<<i+1<<endl;

cout<<"Stambuk: "<<data[i].stambuk<<endl;

cout<<"Nama: "<<data[i].nama<<endl;

cout<<"Nilai Tugas Mandiri: "<<data[i].nilaiTugasMandiri<<endl;

cout<<"Nilai Tugas Proyek: "<<data[i].nilaiTugasProyek<<endl;

cout<<"Nilai UTS: "<<data[i].nilaiUTS<<endl;

cout<<"Nilai UAS: "<<data[i].nilaiUAS<<endl;

cout<<"Nilai Kehadiran: "<<data[i].nilaiKehadiran<<endl;

cout<<"Nilai Akhir(total): "<<data[i].nilaiAkhir<<endl;

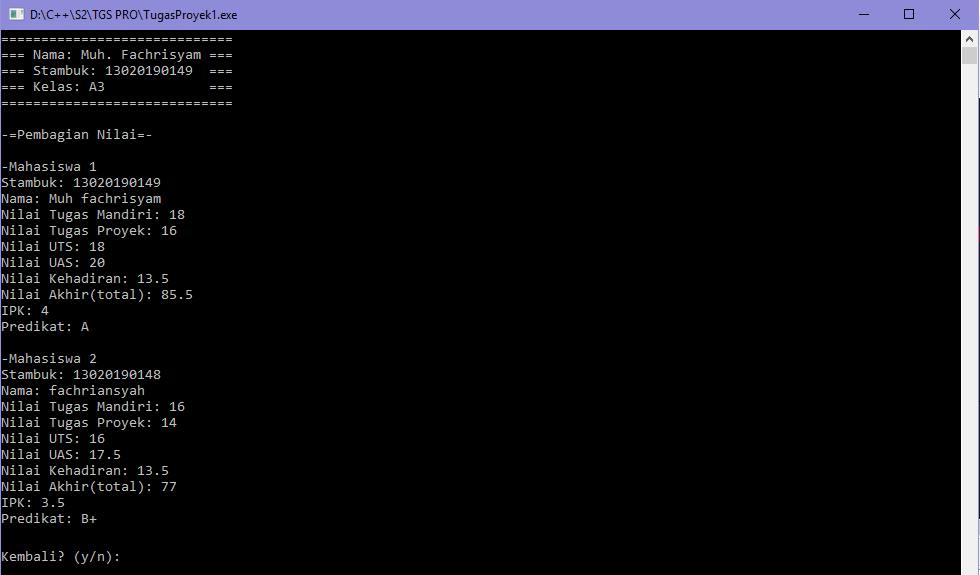
cout<<"IPK: "<<data[i].ipk<<endl;

cout<<"Predikat: "<<data[i].predikat<<endl;

}

}

Akhir dari fungsi search.



Sebelum keluar dari case 2 diberikan pilihan untuk kembali atau tidak dengan memencet “y/n” pada keyboard. Jika menekan ‘y’ maka program akan kembali ke “MENU:” , jika tidak maka program akan selesai

* Dengan menginput angka ‘3’, maka akan dipanggil fungsi “cetakPembagianNilai(mhs,n)” semua data mahasiswa dan jumlah arraynya sebagai parameter. Fungsi ini berfungsi untuk menampilkan semua komponen struct DataMhs

void cetakPembagianNilai(DataMhs data[],int &banyak){

profil();

cout<<"\n-=Pembagian Nilai=-"<<endl;

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

cout<<"\n-Mahasiswa "<<i+1<<endl;

cout<<"Stambuk: "<<data[i].stambuk<<endl;

cout<<"Nama: "<<data[i].nama<<endl;

cout<<"Nilai Tugas Mandiri: "<<data[i].nilaiTugasMandiri<<endl;

cout<<"Nilai Tugas Proyek: "<<data[i].nilaiTugasProyek<<endl;

cout<<"Nilai UTS: "<<data[i].nilaiUTS<<endl;

cout<<"Nilai UAS: "<<data[i].nilaiUAS<<endl;

cout<<"Nilai Kehadiran: "<<data[i].nilaiKehadiran<<endl;

cout<<"Nilai Akhir(total): "<<data[i].nilaiAkhir<<endl;

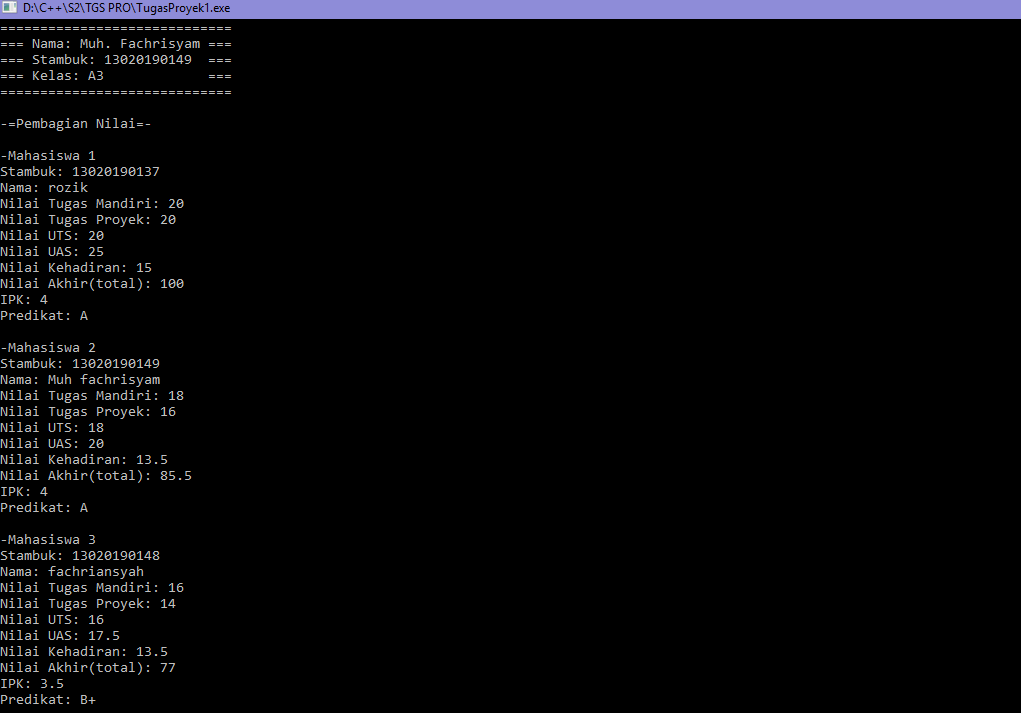
cout<<"IPK: "<<data[i].ipk<<endl;

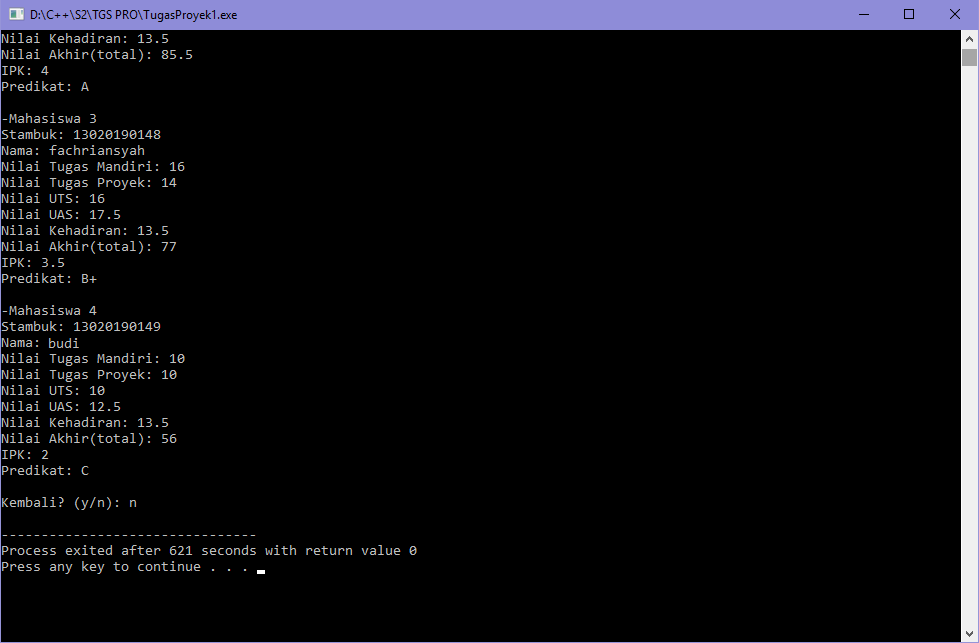
cout<<"Predikat: "<<data[i].predikat<<endl;

}

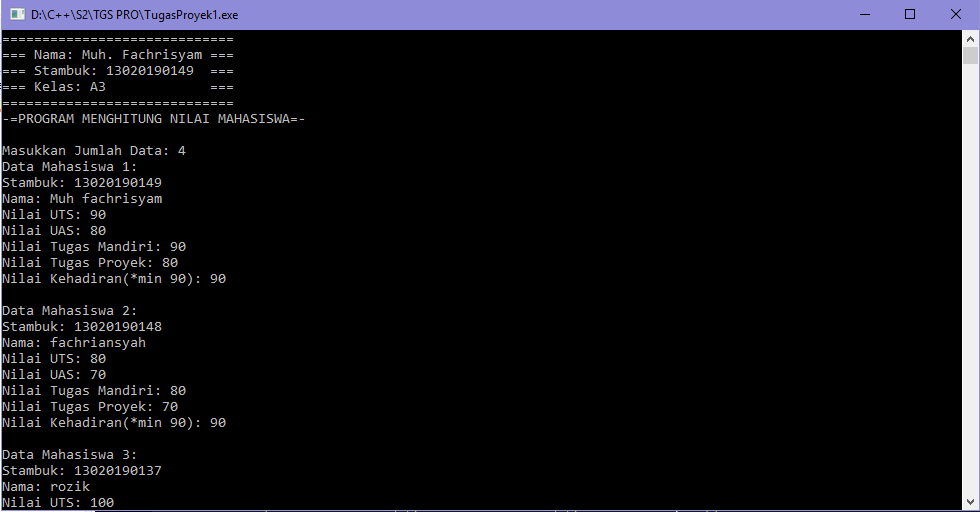
}

Sebelum keluar dari case 2 diberikan pilihan untuk kembali atau tidak dengan memencet “y/n” pada keyboard. Jika menekan “y” maka program akan kembali ke “MENU:” , jika tidak maka program akan selesai

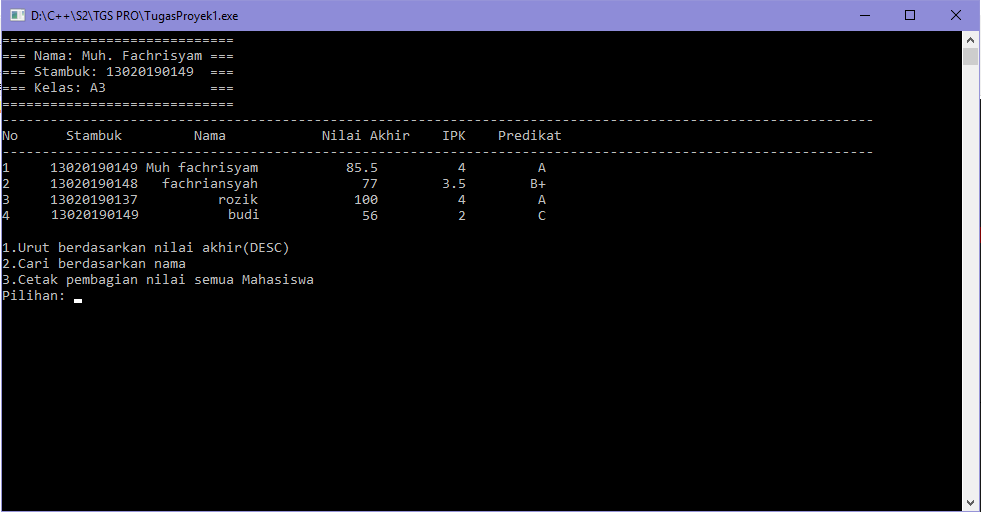




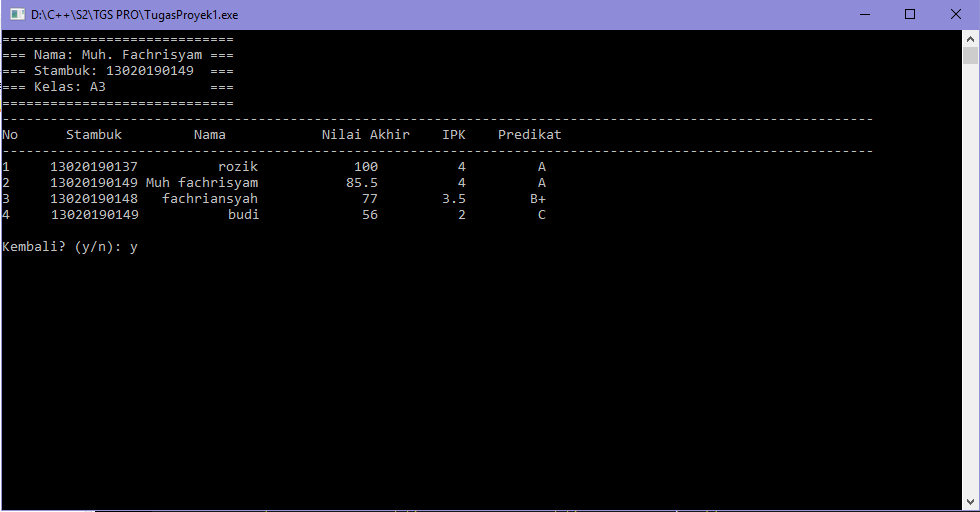
**D. Pengujian Program**

****

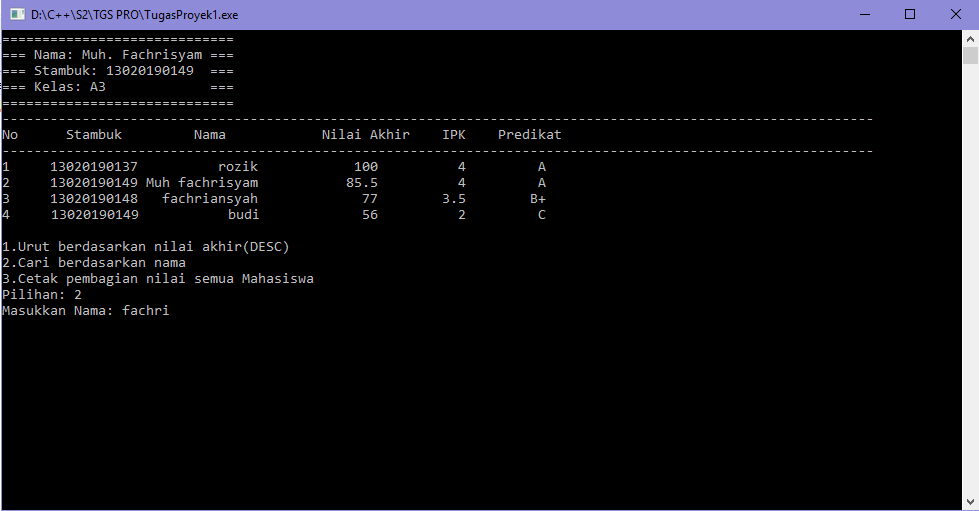
Tampilan awal menginput jumlah data sekaligus menginput setiap nilai mahasiswa

****

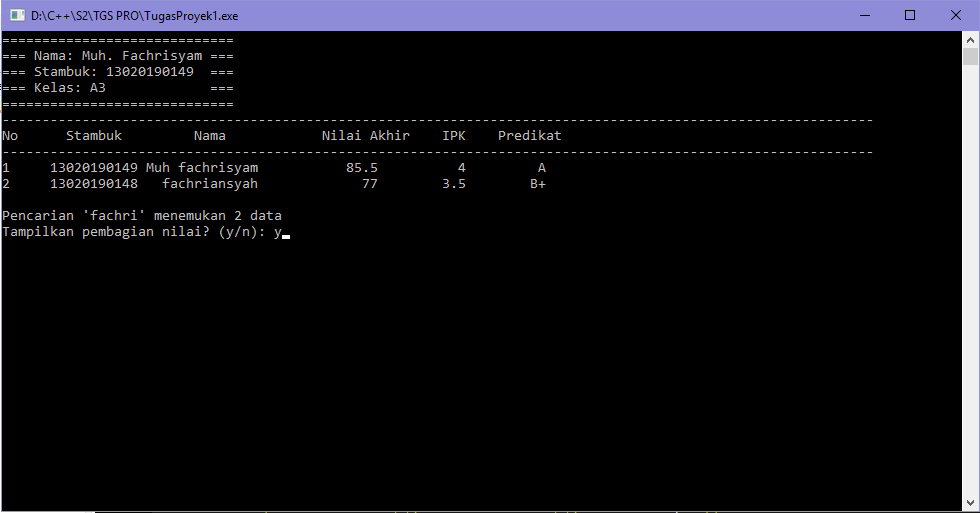
Menampilkan nama,stambuk,nilai akhir,ipk,predikat dari data yang diinput dan perhitungan yang dilakukan dalam bentuk barisan. Kemudian ada menu untuk mengurut, mencari, dan menampilkan pembagian nilai semua mahasiswa.

****

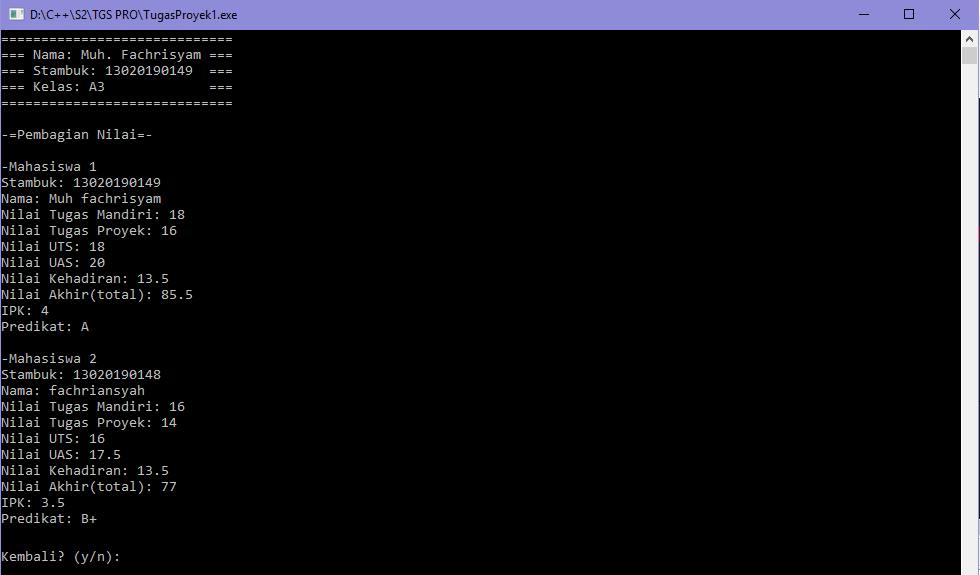
Tampilan setelah menginput 1 yaitu mengurutkan data secara descending, dan ada pilihan untuk kembali ke menu

****

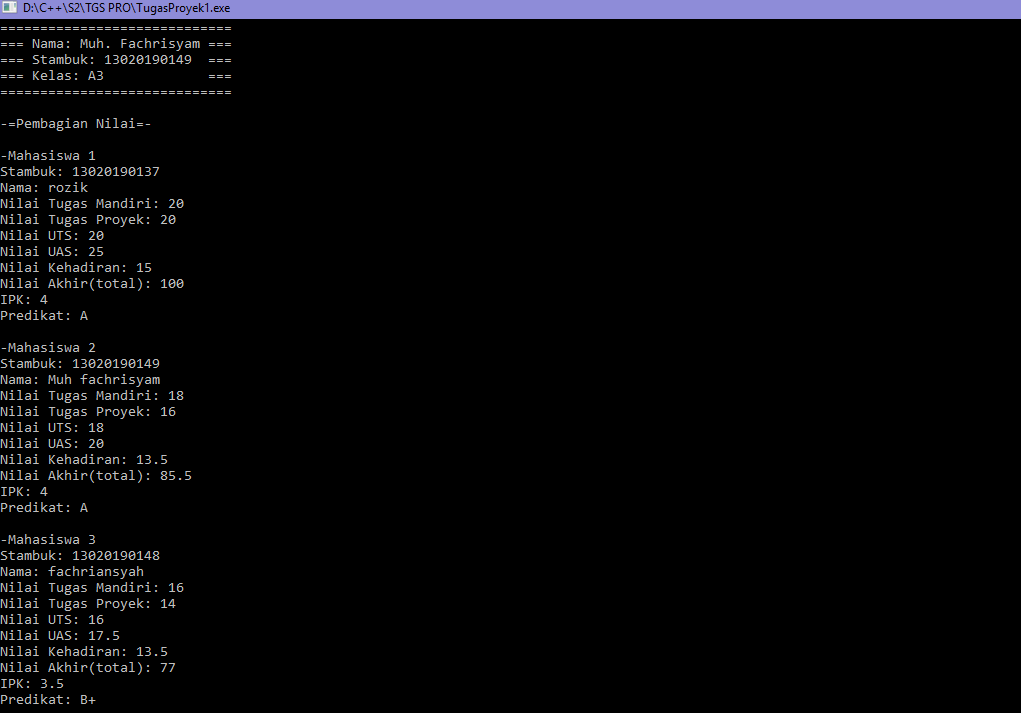
Dengan menginput 2 pada menu, maka akan disuruh untuk menginput nama yang ingin dicari

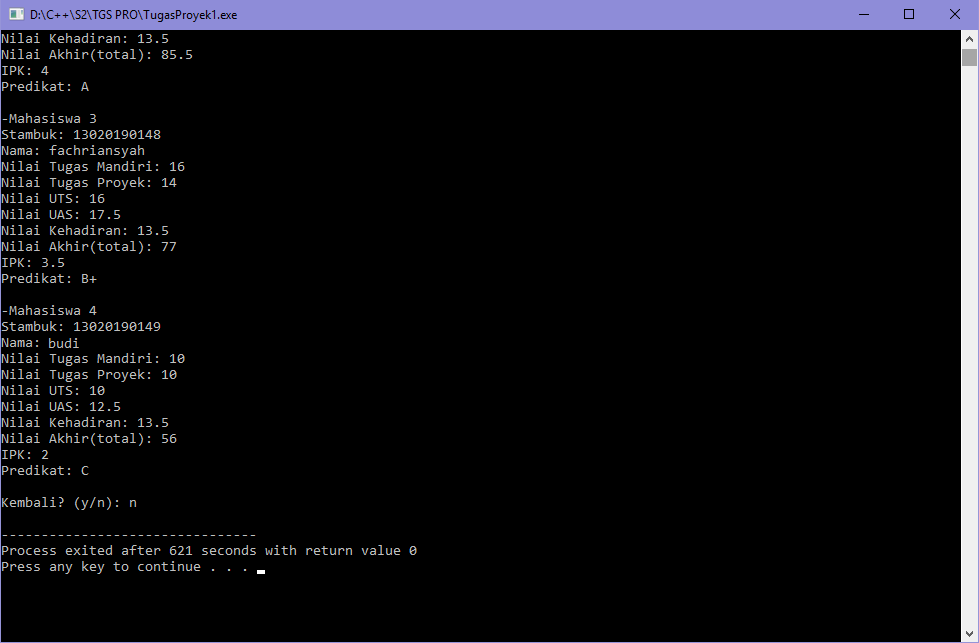
****

Tampilan setelah menemukan data, dan ada pilihan untuk menampilkan pembagian nilai dari mahasiswa pada pencarian

****

Berikut tampilan pembagian nilai dari hasil pencarian. Dan ada pilihan untuk kembali ke menu

****

****

Menginput 3 pada menu akan menampilkan pembagian nilai semua mahasiswa. Dan ada pilihan untuk kembali ke menu. Jika tidak maka program akan selesai

**Penutup**

**A. Kesimpulan dan Saran**

Dalam menyusun program terdapat beberapa hal yang harus di perhatikan mulai dari penyusunan algoritma kemudian penyajiannya. Apabila teks algoritma telah selesai maka teks algoritma tersebut ditranslasi ke program dengan menggunakan bahasa C++ dan aplikasi Dev C++ sebagai compiler. Dalam C++ alur perjalanan fungsi utama dijalankan secara procedural dan terstruktur. Program “Menghitung Nilai Akhir Mahasiswa” hanyalah program sederhana. Untuk pembuatan program yang lebih besar lagi harus mempelajarinya dengan jelas dan baik. Dan juga Untuk mendalami program harus lebih giat dan rajin lagi untuk mengkaji ilmu pemrograman dari berbagai sumber.

**Daftar Pustaka**

https://www.geeksforgeeks.org/sort-c-stl/

https://en.cppreference.com/w/cpp/algorithm/sort

https://stackoverflow.com/questions/9438209/for-every-character-in-string

http://www.cplusplus.com/reference/cstring/strcmp/

https://www.geeksforgeeks.org/structure-sorting-in-c/

https://www.geeksforgeeks.org/string-find-in-cpp/

https://stackoverflow.com/questions/19390059/program-skips-second-cin

http://www.cplusplus.com/reference/string/string/find/

https://en.cppreference.com/w/cpp/named\_req/Compare

https://stackoverflow.com/questions/873715/c-sort-with-structs

https://pascaldhika.blogspot.com/2014/03/macam-macam-file-header-dan-fungsinya\_22.html