1. Input Validasi Menggunakan Loop.

#include <iostream>

using namespace std;

void main(){

int score;

cout << "Input your test score ? ";

cin >> score;

while(score < 0){

cout << "ERROR: the score cannot be less than 0.\n";

cout << "Input the correct score ? ";

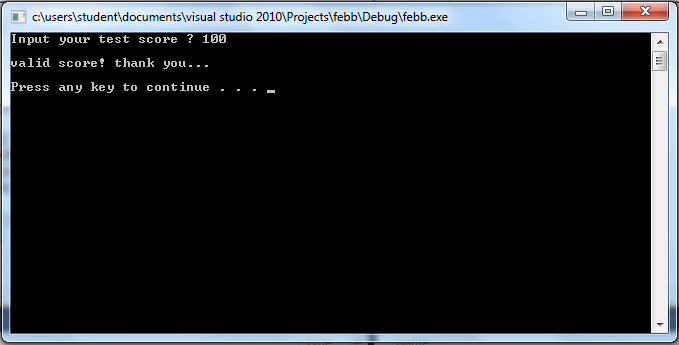
cin >> score;

}

cout << "\nvalid score! thank you...\n\n";

system("pause");

}



1. Validasi Menggunakan Custom Function.

#include <iostream>

using namespace std;

bool isInvalid(int model);

void main(){

int model;

cout << "Input model number ? ";

cin >> model;

while(isInvalid(model)){

cout << "ERROR: The valid model numbers are 100, 200 and 300.\n";

cout << "Input a valid model number ? ";

cin >> model;

}

cout << "\nValid model! Thank you... \n\n";

system("pause");

}

bool isInvalid(int model){

bool status;

if(model != 100 && model != 200 && model != 300){

status = true;

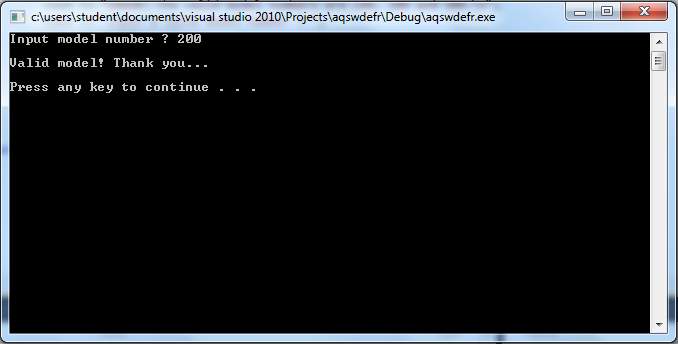
}else{

status = false;

};

return status;

}



1. Validasi String Input.

#include <iostream>

#include <string>

using namespace std;

bool isInvalid(int model);

void main(){

string answer;

cout << "Do you love me ? ";

cin >> answer;

while(answer != "yes"){

cout << "ERROR: The valid answer is 'yes'\n\n";

cout << "Do you love me ? ";

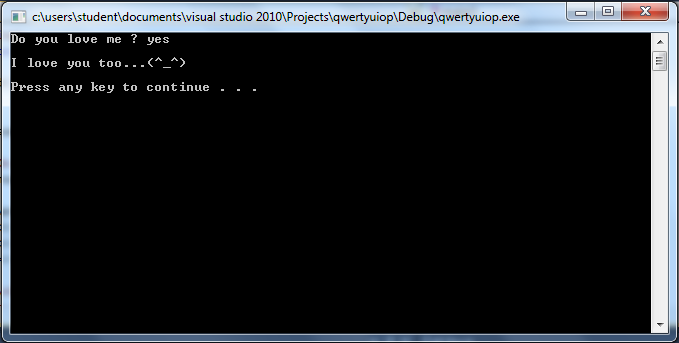
cin >> answer;

}

cout << "\nI love you too...(^\_^)\n\n";

system("pause");

}



1. Tugas Praktikum 5

#include <iostream>

#include <string>

using namespace std;

int angka, a, n, besar=0, kecil=20;

string cobalagi;

void main(){

do{

system("CLS");

cout << " Input banyak bilangan ? ";

cin >> angka;

for(a = 1;a <=angka;a++){

cout << "Input bilangan ke " << a << "?";

cin >> n;

while(n<10 || n>20){

cout << "ERROR: valid range [10-20].\n";

cout << "Input a valid model number ? ";

cin >> n;

}

if (besar < n ) besar = n ;

if (kecil > n ) kecil = n ;

}

cout << "\n\n";

cout << "Bilangan terbesar adalah " << besar << "\n\n";

cout << "Bilangan terkecil adalah " << kecil << "\n\n";

cout << "Coba lagi ? (y/t)";

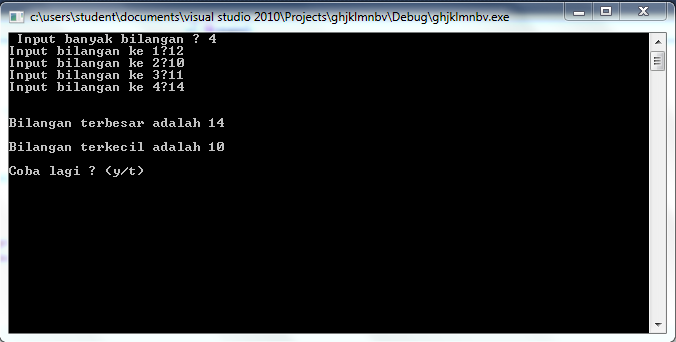
cin >> cobalagi;

}while(cobalagi=="y");

cout << "\n \* Terima kasih sudah menggunakan program ini ....\* ";

system ("pause");

}



Kesimpulan :

Dalam lab kali ini kita dapat memahami apa yang dimaksud dengan Garbage In, dan Garbage Out. Kita juga memahami program error trap yang menvalidasi proses input data sebaga bagian dari defensive programming.