**BAB VI**

**Pengenalan C++**

|  |  |  |
| --- | --- | --- |
| **Nama** | **:** | Sabastian Rafarifza Wahyuono |
| **NIM** | **:** | 2318043 |
| **Tujuan Praktikum** | **:** | 1. Mampu memahami struktur perulangan While 2. Mampu memahami struktur perulangan Do While 3. Mampu membuat program sederhana dari pemahaman sebelumnya |
| **Alat / bahan** | **:** | 1. Seperangkat *computer*. 2. Perangkat lunak: Dev C++. 3. Modul Praktikum Algoritma dan Pemrogramman 2023. |

1. **Landasan Teori**

Struktur perulangan *WHILE* adalah pengulangan yang melakukan pemeriksaan kondisi di awal blok struktur. Pengulangan hanya akan dilakukan jika kondisi yang didefinisikan di dalamnya terpenuhi (bernilai benar). Hal ini berarti jika kondisi yang didefinisikan tidak terpenuhi (bernilai salah) maka statemen-statemen yang terdapat dalam blok pengulangan pun tidak akan pernah dieksekusi oleh program.

Berbeda dengan struktur *while* yang melakukan pemeriksaan kondisi di awal blok perulangan, pada struktur *do*-*while* kondisi justru ditempatkan di bagian akhir. Hal ini tentu menyebabkan struktur pengulangan ini minimal akan **melakukan satu kali** proses walaupun kondisi yang didefinisikan tidak terpenuhi (bernilai salah).

1. **Langkah – Langkah Praktikum**
2. Buka Aplikasi Dev C++.
3. Buka Modul Praktikum Algoritma dan Pemrogramman.
4. Menjalankan script dari modul
5. **Latihan Praktikum ke-1:**
6. Source Code

|  |
| --- |
| #include<iostream>  using namespace std;  main()  {  int i, a=0, b=1, c, n;  cout << "masukan angka : "; cin>>n;  cout<<a<<" "<<b<<" ";  while(i<=n-3){  c=a+b;  a=b;  b=c;  cout<<c<<" ";  i++;  }  } |

1. Hasil Running



Gambar 6. 1 Hasil Angka Fibionic

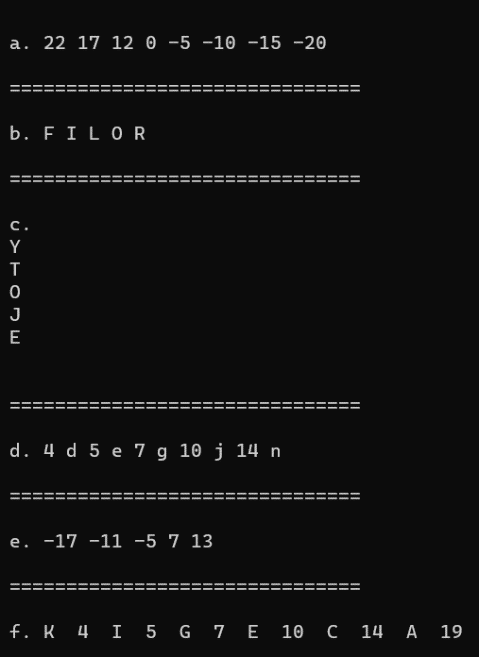
1. Analisa

Progam diatas merupakan progam untuk mencetak angka *fibionic* yang menggunakan tipe data *integer* dengan variabel i, a=0, b=1, c, n. kemudian mengeluarkan perintah input variabel n dan mengeluarkan variabel a dan b. setelah itu menggunakan perulanagan *while* dengan progam i<=n-3 yang didalamnya menggunakan progam c = a + b; a = b; b = c; cout << c << " "; dan i++;

1. **Latihan Praktikum ke-2:**
2. Source Code

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  cout << "\na. ";  int a = 22;  do{  cout << a << " ";  if(a == 12){  a -= 12;  continue;  }  a -= 5;  }  while(a >= -20);  cout <<"\n\n===============================\n\n";  cout << "b. ";  char hurufb = 'F';  do{  cout << hurufb << " ";  hurufb += 3;  }  while(hurufb <= 'R');  cout <<"\n\n===============================\n\n";  cout << "c. \n";  char hurufc = 'Y';  do {  cout<<hurufc<<endl;  hurufc -= 5;  }  while(hurufc >= 'A');  cout <<"\n\n===============================\n\n";  cout << "d. ";  int loop = 0, angka = 4;  char hurufd = 'd';  do {  cout << angka << " ";  cout << hurufd << " ";  loop++;  angka += loop;  hurufd += loop;  }  while(loop < 5);  cout <<"\n\n===============================\n\n";  cout << "e. ";  int keluaran = -17, looper = 0;  do {  cout << keluaran << " ";  if(looper == 2)  {  keluaran += 12;  looper++;  continue;  }  keluaran += 6;  looper++;  }  while(keluaran <= 13);  cout <<"\n\n===============================\n\n";  cout << "f. ";  int well = 4, loopnya = 0;  char done = 'K';  do {  cout << done << " ";  cout << well << " ";  loopnya++;  well += loopnya;  done -= 2;  }  while(loopnya<=5);  cout <<"\n";  } |

1. Hasil Running



Gambar 6. 2 Hasil Ltihan Progam Dari 2

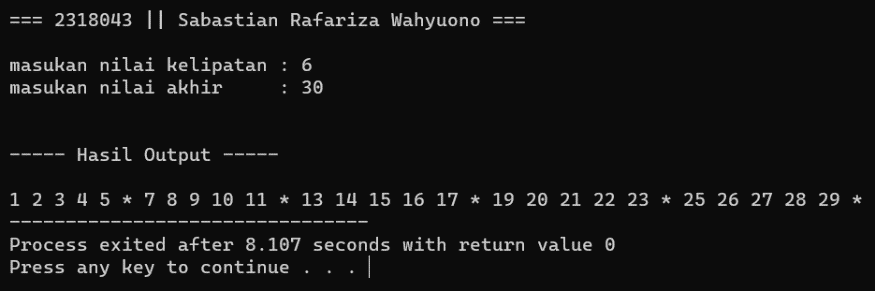
1. Analisa

Progam pertama akan mencetak serangkaian angka dari 22 hingga -20 dengan langkah pengurangan sebesar 5 yang menggunakan perulanagan Do-While. Progam kedua mencetak serangkaian karakter dimulai dari 'F' dan ditambahkan 3 setiap kali melalui *loop* do-while. Progam krtiga mencetak serangkaian karakter dimulai dari 'Y' dan dikurangi 5 setiap kali melalui *loop* do-while hingga huruf 'E'. progam keempat mencetak perulangan angka 4 di tambah nilai perulangan 1 sampai 3, kemudian dilanjutkan dangan serangkaian karakter dimulai dari 'd' dan ditambahkan perulangan 1 sampai 3 setiap kali melalui loop do-while. Kelima menggunakan *loop* do-while untuk mencetak serangkaian angka berdasarkan perubahan nilai variabel keluaran dan *looper*. Keenam hampir sama dengan progam nomer empat

1. **Tugas Praktikum ke-1:**
2. Source Code

|  |
| --- |
| #include<iostream>  using namespace std;  main(){  int kelipatan, nilai\_akhir, i = 1, hasil;  cout << "=== 2318043 || Sabastian Rafariza Wahyuono ===" << endl;  cout << "\nmasukan nilai kelipatan\t: "; cin>>kelipatan;  cout << "masukan nilai akhir\t: "; cin>>nilai\_akhir;  cout << "\n\n----- Hasil Output -----" << endl << endl;  while(i <= nilai\_akhir){  if (i % kelipatan == 0){  cout<<"\* ";  }  else{  cout << i << " ";  }  i++;  }  } |

1. Hasil Running



Gambar 6. 3 Hasi *Run* Tugas Ke 1

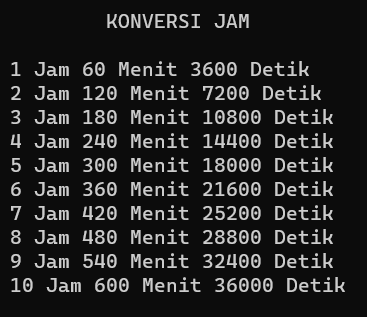
1. Analisa

Progam diatas menggunakan tipedata *integer* dengan variabel kelipatan, nilai\_akhir, i = 1, hasil. Pertama menggunakan perintah input untuk variabel kelipatan dan nilai\_akhir. Kemudian menggunakan perulangan while dengan variabel i <= nilai\_akhir dengan yang diulang menggunakan percabangan if i % kelipatan == 0 kemudian akan mengeluarkan "\* " dan jika tidak atau else akan mengeluarkan variabel i

1. **Tugas Praktikum ke-2:**
2. Source Code

|  |
| --- |
| #include<iostream>  using namespace std;  int main() {  int jam = 1, menit = 60, m = 0, detik = 3600, d = 0;  cout<<"\tKONVERSI JAM"<<endl<<endl;  do{  m = m + menit;  d = d + detik;  cout<<jam<<" Jam "<<m<<" Menit "<<d<<" Detik"<<endl;  jam++;  }  while(jam<=10);  } |

1. Hasil Running



Gambar 6. 4 Hasil Run Tugas 2 Konversi Waktu

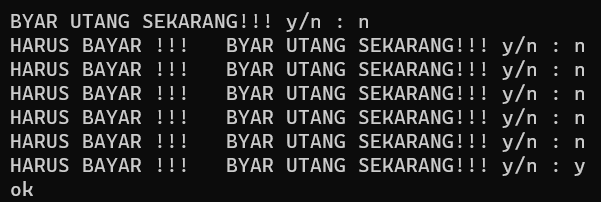
1. Analisa

Progam diatas menggunakan *tipedata* integer dengen variabel int jam = 1, menit = 60, m = 0, detik = 3600, d = 0 dengan rumus masing masing jam menggunakan nili perulangan 1 sampai 10, untuk merubah jam ke menit tinggal menjumblahkan saja dengan 60 pada masing-masing perulangan, untuk mencari detik pun juga begitu.

1. **Tugas praktikum ke-3**
2. Source Code

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  char bayar;  do{  cout<<"BYAR UTANG SEKARANG!!! y/n : ";cin>>bayar;  if (bayar =='y'){cout<<"ok";}  else{cout<<"HARUS BAYAR !!! ";}  }  while(bayar != 'y');  return 0;  } |

1. Hasil Running



Gambar 6. 5 Hasil Progam Untuk Harus Bayah Hutang

1. Analisa

Program ini adalah program C++ yang menggunakan *loop* do-while untuk terus meminta pembayaran utang sampai pengguna memasukkan karakter 'y'. Pertama meminta pengguna untuk memasukkan karakter 'y' atau 'n' untuk menentukan apakah pembayaran utang akan dilakukan atau tidak. Jika pengguna memasukkan 'y', program mencetak "ok". Jika tidak, program mencetak "HARUS BAYAR !!!"

1. **Kesimpulan**
2. Struktur while memeriksa kondisi di awal, sementara do-while memeriksa kondisi di akhir.
3. Pengulangan hanya dilakukan jika kondisi yang didefinisikan di awal blok perulangan terpenuhi (bernilai benar) pada while sedangkan struktur perulangan do-while melakukan pemeriksaan kondisi di akhir blok perulangan.
4. Kedua struktur perulangan ini berguna dalam situasi di mana kita ingin melakukan sesuatu setidaknya satu kali atau berulang sampai kondisi tertentu tidak terpenuhi lagi.