

NAMA : MUHAMMAD LINTANG PAMUNGKAS

PRODI : TEKNIK INFORMATIKA

NIM : 210103028

WAKTU : 08 November 2021

KELAS : TI-A1

1. - Algoritma adalah prosedur komputasi yang terdefinisi dengan baik yang terdiri dari satu set instruksi, yang mengambil beberapa nilai atau set nilai sebagai input, dan menghasilkan beberapa nilai atau set nilai sebagai output. Sedangkan pemrograman merupakan suatu proses coding atau membuat program.
 - Karena algoritma merupakan prosedur (Langkah langkah) penyelesaian masalah dalam bentuk yang mudah dipahami nalar manusia sebagai acuan yang membantu dalam mengembangkan program komputer dalam melakukan pemrograman jadi calon programmer wajib mempelajarinya.
2. - Diagram Alur atau FlowChart, Bagan/diagram yang memiliki aliran yang menggambarkan langkah-langkah penyelesaian masalah
 - Kode Semu (Pseudo Code) : Pseudo code adalah urutan baris algoritma seperti code pemrograman dan tidak memiliki sintak yang baku.
 - Algoritma Fundamental : menyatakan 5 komponen utama dalam algoritma yaitu finiteness, definiteness, input, output, dan effectiveness.
3. - Kalimat Deskriptif : Dengan notasi kalimat deskriptif ini, deskripsi setiap langkah dijelaskan dengan bahasa yang jelas/gamblang
 - Diagram Alir (Flowchart) : Dengan diagram alir, deskripsi setiap langkah dijelaskan dalam sebuah bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses dalam suatu program
 - Pseudo-code : Dengan notasi pseudo-code, deskripsi setiap langkah dijelaskan dengan menggunakan vocabulary (baku) atau dalam arti lain seperti ini, pseudocode merupakan deskripsi tingkat tinggi informal dan ringkas atas algoritma pemrograman komputer yang menggunakan konvensi struktural atas suatu bahasa pemrograman, dan ditujukan untuk dibaca oleh manusia dan bukan oleh mesin.
4. - Bahasa Tingkat Rendah, atau dikenal dengan istilah bahasa rakitan (bah. Inggris Assembly), yaitu memberikan perintah kepada komputer dengan memakai kode-kode singkat (kode mnemonic), contohnya kode_mesin|MOV, SUB, CMP, JMP, JGE, JL, LOOP, dsb.
 - Bahasa Tingkat Menengah, yaitu Bahasa tingkat menengah memberikan satu tingkat abstraksi di atas kode mesin. Bahasa assembly memiliki sedikit semantik atau spesifikasi formal, karena hanya pemetaan simbol yang dapat di baca manusia. Biasanya, satu instruksi mesin diwakili sebagai satu baris kode *assembly*. Assembler menghasilkan file objek yang bisa dihubungkan dengan file objek lain atau dimuat sendiri. CONTOH BAHASA MENENGAH yaitu Assembler dan Microsoft Macro Assembler (MASM)
 - Bahasa Tingkat Tinggi, yaitu bahasa komputer yang memakai instruksi berasal dari unsur kata-kata bahasa manusia, contohnya begin, end, if, for, while, and, or, dsb.[butuh rujukan] Komputer dapat mengerti bahasa manusia itu diperlukan program compiler atau interpreter.
5. Fungsi library untuk mengambil perintah coding atau untuk mendefinisikan berbagai file makro, fungsi, variabel dan konstanta dari pustaka(library). Bentuknya #include<file header>.
 - Contoh <iostream.h> digunakan untuk memanggil endl cout cin. Jadi tanpa header tersebut maka coding tidak akan bisa terbaca.

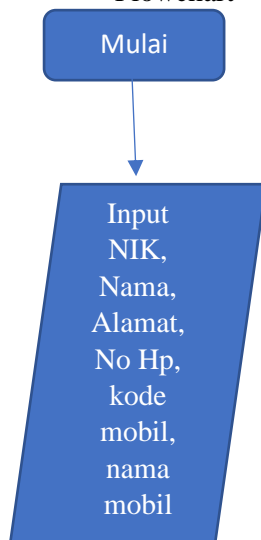
6. Agar program dapat mengenali variable yang akan digunakan untuk memprogram/codingan
7. Karena dalam sebuah coding terdapat pengelompokan suatu tipe variabel yang berbeda maka untuk membedakannya data variabelnya. contoh seperti berikut : integer, string, char, float,dll.

8.

A. Bahasa natural

1. Mulai
2. Input NIK, Nama, Alamat, No Hp, kode mobil, nama mobil
3. Jika > 1 hari maka di kenakan denda 20.000
4. Jika < 1 hari maka denda = 0
5. Cetak biaya mobil, denda dan total bayar = (biaya mobil+denda)
6. Selesai

Flowchart



B. Coding

```
// memilih mobil
// sopir atau tidak
// Sewa berapa hari

// Aturan :
// Jika melebihi batas waktu, denda 20.000/jam

#include <iostream>
#include <string>

using namespace std;

int jumlahPelanggan, lamaRental;
double denda;
char pilihMobil, isSopir;

struct data {
    string nik, nama, alamat;
    int noHP;
};

data pelanggan[5];

void cetak(int jumlahPelanggan, string nik, string nama, string alamat, int biayaMobil, double
denda) {

    int totalBayar = biayaMobil + denda;

    for ( int i = 1; i <= jumlahPelanggan; i++)
    {
        cout << "Data Pelanggan " << i << "\n";
        // cout << "Mobil yang akan dirental " << " ";
    }
}
```

```

        cout << "NIK          : "<< nik[i] <<" \n";
        cout << "Nama          : "<< nama[i] <<" \n";
        cout << "Alamat        : "<< alamat[i] <<" \n";
        cout << "Biaya Mobil   : Rp "<< biayaMobil <<" \n";
        cout << "Denda          : Rp "<< denda <<" \n";
        cout << "Total Biaya   : Rp "<< totalBayar <<" \n";
    }

}

```

```

cekDenda(int lamaRental) {
    if ( lamaRental >= 5 ) // 5 > 5
    {
        double toMenit = lamaRental / 24;
        denda = 20000 * toMenit;
    }
    else {
        denda = 0;
    }
}

```

```

void kelolaData(int jumlahPelanggan, char pilihMobil, int lamaRental, char isSopir, string nik, string
nama, string alamat) {

```

```

    if ( pilihMobil == 'A' || pilihMobil == 'a') // Avanza
    {
        string namaMobil = "Avanza";
        if (isSopir == 'Y' || isSopir == 'y')
        {
            int biayaMobil = 700000 * lamaRental;

```

```

cekDenda(lamaRental);

cetak(jumlahPelanggan, nik, nama, alamat, biayaMobil, denda);
} else {

    int biayaMobil = 500000 * lamaRental;

    cekDenda(lamaRental);

    cetak(jumlahPelanggan, nik, nama, alamat, biayaMobil, denda);

}

} else if( pilihMobil == 'B' || pilihMobil == 'b' ) {

    string namaMobil = "Ertiga";

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

    {

        int biayaMobil = 800000 * lamaRental;

        cekDenda(lamaRental);

        cetak(jumlahPelanggan, nik, nama, alamat, biayaMobil, denda);

    } else {

        int biayaMobil = 600000 * lamaRental;

        cekDenda(lamaRental);

        cetak(jumlahPelanggan, nik, nama, alamat, biayaMobil, denda);

    }

} else if( pilihMobil == 'C' || pilihMobil == 'c' ) {

    string namaMobil = "Ertiga";

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

    {

        int biayaMobil = 1000000 * lamaRental;

        cekDenda(lamaRental);

        cetak(jumlahPelanggan, nik, nama, alamat, biayaMobil, denda);

    } else {

        int biayaMobil = 700000 * lamaRental;

        cekDenda(lamaRental);

        cetak(jumlahPelanggan, nik, nama, alamat, biayaMobil, denda);

    }

}

}

```

```
}
```

```
int main() {
```

```
    cout << "Ingin mengolah berapa data : ";
```

```
    cin >> jumlahPelanggan;
```

```
    cout << "\n";
```

```
    cout <<
    "+=====+\n";
```

```
    cout << "|          Daftar Mobil          |\n";
```

```
    cout <<
    "+=====+\n";
```

```
    cout << "| A | M-001 |   Avanza   |   500.000   | 700.000 (Sopir)\n";
```

```
    cout <<
    "+=====+\n";
```

```
    cout << "| B | M-002 |   Ertiga   |   600.000   | 800.000 (Sopir)\n";
```

```
    cout <<
    "+=====+\n";
```

```
    cout << "| C | M-003 |  Expander  |   700.000   | 1.000.000 (Sopir)\n";
```

```
    cout <<
    "+=====+\n";
```

```
    cout << "\n";
```

```
    for (int i = 1; i <= jumlahPelanggan; i++)
```

```
    {
```

```
        cout << "Data Pelanggan " << i << "\n";
```

```
        cout << "Mobil yang akan dirental (A / B / C) : ";
```

```
        cin >> pilihMobil;
```

```
        cout << "Pakai sopir / tidak (Y/n) : ";
```

```
        cin >> isSopir;
```

```
cout << "Rental berapa hari? : ";
```

```
cin >> lamaRental;
```

```
cout << "NIK : ";
```

```
cin >> pelanggan[i].nik;
```

```
cout << "Nama : ";
```

```
cin >> pelanggan[i].nama;
```

```
cout << "Alamat : ";
```

```
cin >> pelanggan[i].alamat;
```

```
cout << "NO HP : ";
```

```
cin >> pelanggan[i].noHP;
```

```
cout << "\n";
```

```
    kelolaData(jumlahPelanggan, pilihMobil, lamaRental, isSopir, pelanggan[i].nik.c_str(),  
    pelanggan[i].nama.c_str(), pelanggan[i].alamat.c_str());
```

```
}
```

```
// for (int i = 1; i <= jumlahPelanggan; i++)
```

```
// {
```

```
//     cout << "Nama : "<< pelanggan[i].nama <<"";
```

```
// }
```

```
}
```