

**LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II**



NAMA : SATRIA SEPTA ARIANTO
NIM : 193020503026
KELAS : A
**MODUL : I (DASAR PEMROGRAMAN
BERORIENTASI OBJEK)**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2020**

LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II



Nama : Satria Septa Arianto
NIM : 193020503026
Kelas : A
Modul : Dasar Pemrograman Berorientasi Objek

Komposisi	MAX	Nilai
BAB I Tujuan dan Landasan Teori	10	7
BAB II Pembahasan	60	50
BAB III Kesimpulan	20	10
Daftar Pustaka	5	4
Lampiran	5	3
Jumlah	100	74

Penilai

Asisten Praktikum

Diana

BAB I

TUJUAN DAN LANDASAN TEORI

yus

1.1 TUJUAN

Setelah menyelesaikan modul ini, mahasiswa diharapkan mampu:

1. Memahami dasar-dasar pemrograman berorientasi obyek
2. Memahami enkapsulasi
3. Membuat kelas dan objek

I ← 1 enter

1.2 LANDASAN TEORI

Pemrograman Berorientasi Objek (PBO) atau Object Oriented Programming (OOP) adalah metode pemrograman yang berorientasikan kepada objek, dimana semua data dan fungsi dalam metode ini didefinisikan ke dalam kelas-kelas atau objek-objek agar bisa saling bekerjasama dalam memecahkan masalah. Model data berorientasi objek dapat memberikan fleksibilitas yang lebih banyak, memberikan kemudahan dalam mengubah program, dan bisa digunakan secara luas dalam pemrograman skala besar.

Pemrograman Berorientasi Objek juga merupakan pembungkusan data (attributes) dan fungsi (method) ke paket yang disebut kelas. Attribute adalah data yang menggambarkan status internal sebuah objek. Seperti anggota variabel pada C++. Attribute tidak dapat diakses dari luar karena diberlakukannya enkapsulasi. Enkapsulasi adalah sebuah metoda untuk mengatur struktur class dengan cara menyembunyikan alur kerja dari class tersebut. Struktur class yang dimaksud adalah attribute dan method. Method adalah fungsi yang mengakses status internal sebuah objek. Method inilah yang berfungsi sebagai perantara antara program luar dengan attribute karena method dapat memanipulasi attribute dan juga dapat diakses dari luar. Disebut juga anggota fungsi pada C++. Kelas (Class) terdiri dari model objek yang memiliki attribute (data members) dan method (member

functions). Member functions yaitu methods yang dipanggil sebagai respons terhadap pesan. Kelas didefinisikan dengan keyword class.

Class (Kelas), adalah kerangka dasar dari objek yang akan diciptakan, bisa berupa struktur yang mendefinisikan data atau method dari objek. Contoh penamaan kelas adalah: Motor, Laptop, Anggota, dan lainnya.

Method, adalah perilaku dari sebuah class. Bisa juga disebut sebagai tindakan yang bisa dilakukan oleh suatu class. Contoh pada class Motor, memiliki method sebagai berikut:

1. Start, method untuk menjalankan motor
2. Stop, method untuk menghentikan laju motor
3. Ganti Gigi, method untuk ganti gigi
4. Turn, method untuk belok kiri atau kanan

Variabel Kelas, merupakan variabel yang dibagikan oleh semua turunan dari kelas. Variabel kelas didefinisikan di dalam kelas, tapi di luar method yang ada di dalam kelas tersebut.

Data Member, merupakan variabel penyimpan data yang berhubungan dengan kelas dan objek.

BAB II

PEMBAHASAN

2.1 Program Pertama

```
#include <iostream>
#include <conio.h>

using namespace std;

class Buah
{
    public:
        Buah(char, int);
    private:
        char fruit;
        int no;
};

Buah::Buah(char f, int n)
{
    fruit=f;
    no=n;
}

int main()
{
    int buah;
    int pilih;

    system("cls");
    cout<<" Program Warna Buah Matang"<<endl;
    cout<<" =====\n\n";
    cout<<" 1. Daftar\n";
    cout<<" 2. Keluar\n\n";
    cout<<" Pilih : "; cin>>pilih;
    system("cls");

    switch(pilih)
    {
        case 1:
        {
            cout<<" =====DAFTAR BUAH=====\\n"<<endl;
            cout<<" Fyi, ini hanyalah buah yang terdapat di  
sekitar rumah saya\\n\\n"<<endl;
            cout<<" 1. Jambu Biji"<<endl;
            cout<<" 2. Kersen"<<endl;
            cout<<" 3. Markisa"<<endl;
            cout<<" 4. Mangga"<<endl;
```

```
cout<<" 5. Nanas"<<endl;
cout<<" 6. Pepaya"<<endl;
cout<<" 7. Pisang\n\n"<<endl;
cout<<" Pilih buah : "; cin>>buah;
cout<<" \n";
switch(buah)
{
    case 1:
    {
        cout<<" Jika sudah matang
        berwarna hijau
        kekuningan"<<endl;
        getch();
        break;
    }

    case 2:
    {
        cout<<" Jika sudah matang
        berwarna merah"<<endl;
        getch();
        break;
    }

    case 3:
    {
        cout<<" Jika sudah matang
        berwarna kuning"<<endl;
        getch();
        break;
    }

    case 4:
    {
        cout<<" Jika sudah matang
        berwarna hijau/hijau
        kekuningan"<<endl;
        getch();
        break;
    }

    case 5:
    {
        cout<<" Jika sudah matang
        berwarna
        kuning/oranye"<<endl;
        getch();
        break;
    }
}
```

```

        case 6:
        {
            cout<<" Jika sudah matang
            berwarna hijau
            kekuningan"<<endl;
            getch();
            break;
        }

        case 7:
        {
            cout<<" Jika sudah matang
            berwarna kuning"<<endl;
            getch();
            break;
        }

        default:
        {
            cout<<" Pilihaneun eopseo";
            break;
        }
    };
    getch();
    return 0;
    break;
}

case 2:
{
    cout<<" Annyeong";
    break;
}

default:
{
    cout<<" Pilihaneun eopseo";
    break;
}
};
while(pilih>0 && pilih<2);

return 0;
}

```

PENJELASAN

```
#include <iostream>
#include <conio.h>
```

Merupakan perintah untuk menginclude library yang digunakan dalam program.

```
using namespace std;
```

Merupakan inputan untuk penulisan kode langsung tanpa penulisan scope std::

```
class Buah
{
    public:
        Buah(char, int);
    private:
        char fruit;
        int no;
};

Buah::Buah(char f, int n)
{
    fruit=f;
    no=n;
}
```

Merupakan bagian class dan konstruktor. Bagian terpenting dalam PBO karena merupakan method untuk mengakses attribute di dalamnya.

```
int main()
{
    int buah;
    int pilih;

    system("cls");
    cout<<" Program Warna Buah Matang"<<endl;
    cout<<" =====\n\n";
    cout<<" 1. Daftar\n";
    cout<<" 2. Keluar\n\n";
    cout<<" Pilih : "; cin>>pilih;
    system("cls");
}
```



```

switch(pilih)
{
    case 1:
    {
        cout<<" =====DAFTAR BUAH=====\\n"<<endl;
        cout<<" Fyi, ini hanyalah buah yang terdapat di
        sekitar rumah saya\\n\\n"<<endl;
        cout<<" 1. Jambu Biji"<<endl;
        cout<<" 2. Kersen"<<endl;
        cout<<" 3. Markisa"<<endl;
        cout<<" 4. Mangga"<<endl;
        cout<<" 5. Nanas"<<endl;
        cout<<" 6. Pepaya"<<endl;
        cout<<" 7. Pisang\\n\\n"<<endl;
        cout<<" Pilih buah : "; cin>>buah;
        cout<<" \\n";
        switch(buah)
        {
            case 1:
            {
                cout<<" Jika sudah matang
                berwarna hijau
                kekuningan"<<endl;
                getch();
                break;
            }

            case 2:
            {
                cout<<" Jika sudah matang
                berwarna merah"<<endl;
                getch();
                break;
            }

            case 3:
            {
                cout<<" Jika sudah matang
                berwarna kuning"<<endl;
                getch();
                break;
            }

            case 4:
            {
                cout<<" Jika sudah matang
                berwarna hijau/hijau
                kekuningan"<<endl;
                getch();
            }
        }
    }
}

```

```

        break;
    }

    case 5:
    {
        cout<<" Jika sudah matang
        berwarna
        kuning/oranye"<<endl;
        getch();
        break;
    }

    case 6:
    {
        cout<<" Jika sudah matang
        berwarna hijau
        kekuningan"<<endl;
        getch();
        break;
    }

    case 7:
    {
        cout<<" Jika sudah matang
        berwarna kuning"<<endl;
        getch();
        break;
    }

    default:
    {
        cout<<" Pilihaneun eopseo";
        break;
    }
};
getch();
return 0;
break;
}

case 2:
{
    cout<<" Annyeong";
    break;
}

default:
{
    cout<<" Pilihaneun eopseo";

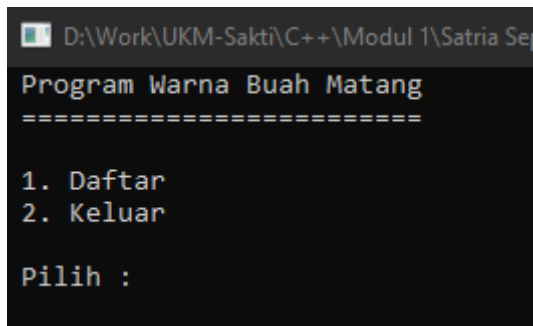
```

```
        break;
    }
};
while(pilih>0 && pilih<2);

return 0;

}
```

Merupakan program utama dimana semua fungsi akan terdapat disini. Untuk class, method yang ada di dalam class akan ditampilkan di dalam sini untuk melakukan proses running sehingga dapat berjalan dengan baik.



```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Sep
Program Warna Buah Matang
=====
1. Daftar
2. Keluar
Pilih :
```

Gambar 2.1 . . .

```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Septa Arianto - Program Buah 1.exe
=====DAFTAR BUAH=====

Fyi, ini hanyalah buah yang terdapat di sekitar rumah saya

1. Jambu Biji
2. Kersen
3. Markisa
4. Mangga
5. Nanas
6. Pepaya
7. Pisang

Pilih buah : 1

Jika sudah matang berwarna hijau kekuningan
```

Gambar 2.2

```
Pilih buah : 2
Jika sudah matang berwarna merah
```

Gambar 2.3

```
Pilih buah : 3
Jika sudah matang berwarna kuning
```

Gambar 2.4

```
Pilih buah : 4
Jika sudah matang berwarna hijau/hijau kekuningan
```

Gambar 2.5

```
Pilih buah : 5  
Jika sudah matang berwarna kuning/oranye
```

Gambar 2.6

```
Pilih buah : 6  
Jika sudah matang berwarna hijau kekuningan
```

Gambar 2.7

```
Pilih buah : 7  
Jika sudah matang berwarna kuning
```

Gambar 2.8

Berdasarkan 8 output diatas diketahui bahwa setiap buah telah disimpan datanya dan inputan pada program hanya untuk memilih buah mana yang akan ditampilkan. Sebagaimana pada Gambar 2.2 adalah daftar buah yang tersedia dan dari Gambar 2.2 sampai Gambar 2.8 adalah output yang dihasilkan dari input yang diberikan.

2.2 Program Kedua

```
#include <iostream>  
  
using namespace std;  
  
class Buah  
{  
    public:  
        Buah(char, char);  
        void input();  
};
```

```

        void output();
    private:
        char nama[20];
        char warna[20];
};

Buah::Buah(char n, char w)
{
    nama[20]=n;
    warna[20]=w;
}

void Buah::input()
{
    cout<<" Masukkan nama buah  : ";cin>>nama;
    cout<<" Masukkan warna buah : ";cin>>warna;
    cout<<" \n"<<endl;
}

void Buah::output()
{
    cout<<" Nama Buah  : "<<nama<<endl;
    cout<<" Warna Buah : "<<warna<<endl;
}

int main()
{
    system("cls");
    cout<<" -----"<<endl;
    cout<<" Input dan Output Buah"<<endl;
    cout<<" -----"<<endl;
    cout<<"\n\n";
    Buah buah(1, 2);
    buah.input();
    buah.output();

    return 0;
}

```

PENJELASAN

```
#include <iostream>
```

Merupakan perintah untuk menginclude library yang digunakan dalam program.

```
using namespace std;
```

Merupakan inputan untuk penulisan kode langsung tanpa penulisan scope std::

```
class Buah
{
    public:
        Buah(char, char);
        void input();
        void output();
    private:
        char nama[20];
        char warna[20];
};

Buah::Buah(char n, char w)
{
    nama[20]=n;
    warna[20]=w;
}

void Buah::input()
{
    cout<<" Masukkan nama buah : ";cin>>nama;
    cout<<" Masukkan warna buah : ";cin>>warna;
    cout<<" \n"<<endl;
}

void Buah::output()
{
    cout<<" Nama Buah : "<<nama<<endl;
    cout<<" Warna Buah : "<<warna<<endl;
}
```

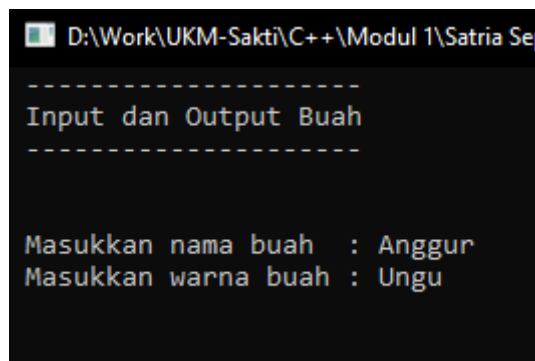
Merupakan bagian class dan konstruktor beserta method dalam bagian class. Bagian terpenting dalam PBO karena merupakan method untuk mengakses attribute di dalamnya. Nantinya akan dipanggil di program utama.

```
int main()
{
    system("cls");
    cout<<" -----"<<endl;
    cout<<" Input dan Output Buah"<<endl;
    cout<<" -----"<<endl;
```

```
    cout<<"\n\n";  
    Buah buah(1, 2);  
    buah.input();  
    buah.output();  
  
    return 0;  
}
```

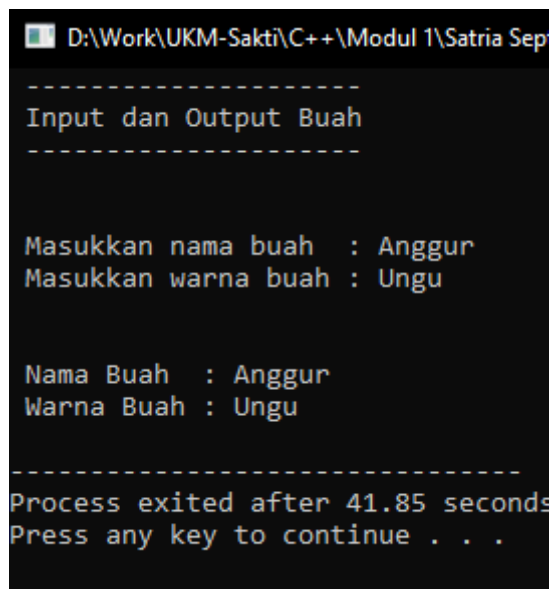
Merupakan program utama dimana semua fungsi akan terdapat disini.

Untuk class, method yang ada di dalam class akan ditampilkan di dalam sini untuk melakukan proses running sehingga dapat berjalan dengan baik.



```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Sep  
-----  
Input dan Output Buah  
-----  
  
Masukkan nama buah : Anggur  
Masukkan warna buah : Ungu
```

Gambar 2.9



```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Sep  
-----  
Input dan Output Buah  
-----  
  
Masukkan nama buah : Anggur  
Masukkan warna buah : Ungu  
  
Nama Buah : Anggur  
Warna Buah : Ungu  
  
-----  
Process exited after 41.85 seconds  
Press any key to continue . . .
```

Gambar 2.10

Berdasarkan 2 output diatas diketahui bahwa input yang diberikan dalam output program langsung ditampilkan pada layar yang sama. Pada Gambar 2.9 adalah input yang diberikan dan hasilnya ada pada Gambar 2.10

BAB III

KESIMPULAN

g.v.

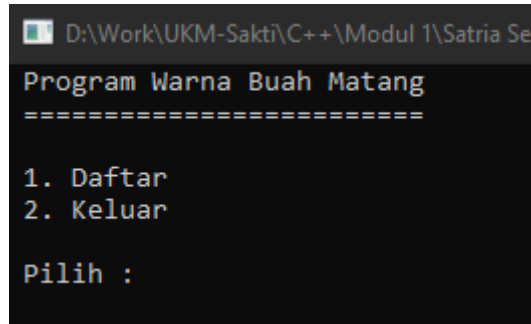
Pemrograman Berorientasi Objek melakukan pembungkusan object yaitu attribute dan method dalam sebuah kelas yang disebut class. Hal ini sangat memudahkan dimana program luar tidak dapat sembarang mengakses variabel yang ada pada class melainkan harus menggunakan method atau fungsi yang tersedia.

•

DAFTAR PUSTAKA

- Dosen Teknik Informatika. Algoritma dan Pemrograman. 2020. *Modul Praktikum Algoritma dan Pemrograman II*. Universitas Palangka Raya. Fakultas Teknik. Jurusan Teknik Informatika.
- Teddy. 2018. *6 Contoh Pemrograman Berorientasi Objek (OOP) untuk Praktek Mandiri*. <https://dosenit.com/kuliah-it/pemrograman/contoh-pemrograman-berorientasi-objek> (Diakses pada : Rabu, 08 Maret 2020 pada pukul : 20.06 WIB)
- Harefa, Jeklin. 2016. *Class and Object : Object Oriented Programming using C++* <https://socs.binus.ac.id/2016/12/13/class-and-object-object-oriented-programming-using-c/> (Diakses pada : Rabu, 08 Maret 2020 pada pukul : 20.03 WIB)

LAMPIRAN

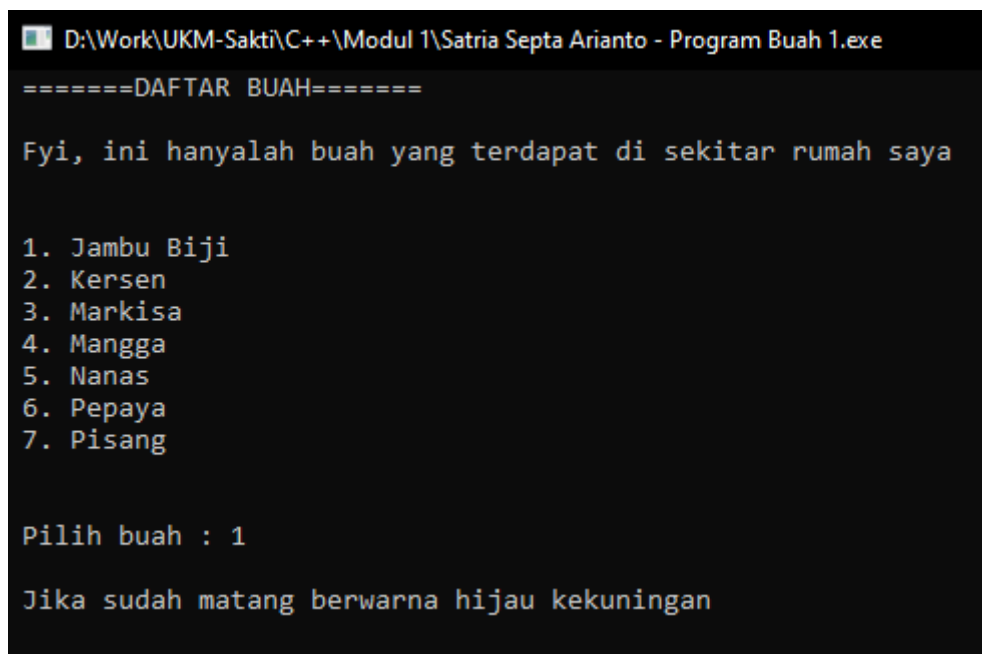


```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Sep
Program Warna Buah Matang
=====

1. Daftar
2. Keluar

Pilih :
```

Gambar 2.1



```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Septa Arianto - Program Buah 1.exe
=====DAFTAR BUAH=====

Fyi, ini hanyalah buah yang terdapat di sekitar rumah saya

1. Jambu Biji
2. Kersen
3. Markisa
4. Mangga
5. Nanas
6. Pepaya
7. Pisang

Pilih buah : 1

Jika sudah matang berwarna hijau kekuningan
```

Gambar 2.2

```
Pilih buah : 2  
Jika sudah matang berwarna merah
```

Gambar 2.3

3 1.5

```
Pilih buah : 3  
Jika sudah matang berwarna kuning
```

Gambar 2.4

3 1 enter

```
Pilih buah : 4  
Jika sudah matang berwarna hijau/hijau kekuningan
```

Gambar 2.5

```
Pilih buah : 5  
Jika sudah matang berwarna kuning/oranye
```

Gambar 2.6

```
Pilih buah : 6  
Jika sudah matang berwarna hijau kekuningan
```

Gambar 2.7

```
Pilih buah : 7  
Jika sudah matang berwarna kuning
```

Gambar 2.8

```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Sep  
-----  
Input dan Output Buah  
-----  
  
Masukkan nama buah : Anggur  
Masukkan warna buah : Ungu
```

Gambar 2.9

```
D:\Work\UKM-Sakti\C++\Modul 1\Satria Sep  
-----  
Input dan Output Buah  
-----  
  
Masukkan nama buah : Anggur  
Masukkan warna buah : Ungu  
  
Nama Buah : Anggur  
Warna Buah : Ungu  
  
-----  
Process exited after 41.85 seconds  
Press any key to continue . . .
```

Gambar 2.10