

**LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II**



NAMA : SATRIA SEPTA ARIANTO
NIM : 193020503026
KELAS : A
MODUL : II (PEWARISAN)

**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 : Pewarisan

Komposisi	MAX	Nilai
BAB I Tujuan dan Landasan Teori	10	8
BAB II Pembahasan	60	50
BAB III Kesimpulan	20	13
Daftar Pustaka	5	5
Lampiran	5	5
Jumlah	100	81

Penilai
Asisten Praktikum

Diana

BAB I

TUJUAN DAN LANDASAN TEORI

1.1 Tujuan

Setelah menyelesaikan modul ini, mahasiswa diharapkan mampu :

- 1.1.1 Mengerti dengan pewarisan
- 1.1.2 Mengerti gagasan tentang kelas dasar dan kelas turunan
- 1.1.3 Membuat kelas baru dari kelas yang sudah ada dengan pewarisan

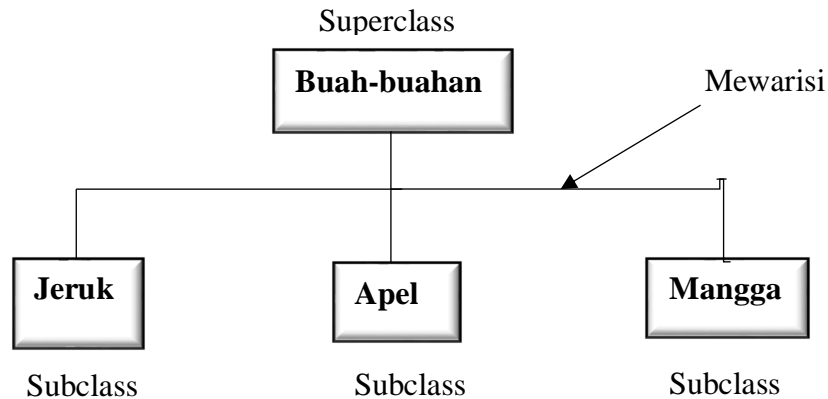
1.2 Landasan Teori

Dalam PBO, kita mengambil realita kehidupan sehari-hari. Kita melakukan pengamatan bahwa manusia secara alami sering melakukan pengelompokan atas objek atau benda. Sejauh ini kita mengetahui cara untuk melakukan pengelompokan – pengelompokan atas objek-objek yang serupa (menjadi kelas objek).

Selain melakukan kategorisasi terhadap objek yang memiliki sekumpulan atribut dan perilaku yang sama, manusia sering melakukan pengelompokan terhadap objek yang memiliki kesamaan atas beberapa (tidak semua) atribut/perilaku. Contoh: Pengelompokan atas buah-buahan, kemudian mengelompokkannya berdasarkan suatu tipe atau jenis (jeruk, apel, mangga, dll.). Setiap subkategori ini merupakan kelas atas objek-objek yang serupa.

Ada beberapa karakteristik yang di-share oleh semua kelompok. Relasi antar kelas-kelas ini disebut dengan relasi “is-a”. Dalam setiap kasus, objek yang dikelompokkan bersama dalam satu sub-kategori merupakan anggota dari kategori yang lebih umum. Contohnya adalah seperti di bawah ini.

- Jeruk adalah (“is-a”) buah-buahan
- Apel adalah (“is-a”) buah-buahan
- Mangga adalah (“is-a”) buah-buahan



Gambar 1.2.1

Objek yang dikelompokkan dalam satu kelas men-share sekumpulan atribut dan perilaku. Jadi, seluruh objek buah-buahan memiliki sekumpulan atribut dan perilaku yang juga dimiliki oleh objek dari jeruk. Keterkaitan antar kelas dalam relasi “is-a” berasal dari kenyataan bahwa sub kelas memiliki atribut dan perilaku yang dimiliki kelas induk, ditambah atribut dan perilaku yang dimiliki oleh sub kelas tersebut.

Superclass (“kelas dasar” atau “kelas induk”) merupakan kelas yang lebih general dalam relasi “is-a”. *Subclass* (“kelas turunan” atau “kelas anak”) merupakan kelas yang lebih spesifik dalam relasi “is-a”. Objek yang dikelompokkan dalam sub kelas memiliki atribut dan perilaku kelas induk, dan juga atribut dan perilaku tambahan. (Jadi, kumpulan atribut dan perilaku sub kelas lebih besar dari super kelas-nya). Relasi “is-a” antar *superclass* dan *subclasses*-nya disebut dengan *pewarisan* atau *inheritance*.

Inheritance atau *Pewarisan/Penurunan* adalah konsep pemrograman dimana sebuah *class* dapat menurunkan *property* dan *method* yang dimilikinya kepada *class* lain. Konsep *inheritance* digunakan untuk memanfaatkan fitur ‘code reuse’ untuk menghindari duplikasi kode program.

Dalam kelas pada pemrograman C++, terdapat mekanisme keturunan (*inheritance*), dimana dalam mekanisme ini terdapat istilah kelas induk dan

kelas anak. Kelas anak akan menginduk pada kelas induk. Keistimewaan dari metode *inheritance* adalah, kelas anak memiliki sifat-sifat atau fungsi dari kelas induk, namun dengan “batasan” tertentu. Dengan mekanisme keturunan ini, member kelas anak akan bertambah yaitu member kelas anak sendiri ditambah dengan member kelas induk. Konsep pewarisan akan mempermudah kita dalam membuat dan perawatan sebuah program, dengan menghilangkan “penggunaan kode berulang” saat pembuatan banyak *class* yang serupa.

Konsep *inheritance* membuat sebuah struktur atau ‘hierarchy’ class dalam kode program. *Class* yang akan ‘diturunkan’ bisa disebut sebagai *class* induk (*parent class*), *super class*, atau *base class*. Sedangkan *class* yang ‘menerima penurunan’ bisa disebut sebagai *class* anak (*child class*), *sub class*, *derived class* atau *heir class*.

Tidak semua *property* dan *method* dari *class* induk akan diturunkan. *Property* dan *method* dengan hak akses *private*, tidak akan diturunkan kepada *class* anak. Hanya *property* dan *method* dengan hak akses *protected* dan *public* saja yang bisa diakses dari *class* anak.

BAB II

PEMBAHASAN

2.1 Program Pewarisan Buah Jeruk

Program pewarisan pertama adalah program pewarisan dengan buah jeruk sebagai objeknya karena masih berkaitan dengan Pemrograman Berorientasi Objek (PBO).

```
#include <iostream>
#include <string>
```

Pada bagian awal, adalah dimana *library* yang diperlukan akan diberi perintah *include*. *Include* berarti memasukkan *library* di sebelahnya. *Iostream* merupakan *library* yang memiliki sintaks yang paling sering digunakan yaitu *cin* dan *cout*. *String* adalah *library* yang khusus menampung tipe data *string*. Tanpa memanggil *library string* maka tipe data *string* tidak dapat digunakan.

```
using namespace std;
```

Using namespace std untuk memudahkan pemanggilan kata perintah tanpa embel-embel *std::*.

```
class buah
{
    protected:
        string nama;
    public:
        buah(string nm): nama()
        {
            nama = nm;
        }
};
```

Merupakan kelas induk/kelas basis/*superclass* dalam program ini. Memiliki satu konstruktor pada bagian *public* dan satu variabel pada bagian *protected*. Variabel inilah yang nantinya akan diwariskan pada kelas anak.

```
class jeruk : public buah
{
    private:
        string warna;
        string latin;
    public:
        jeruk(string nm, string wrn, string lt ) : nama(nm), warna(wrn), latin(lt)
        {
            warna = wrn;
            latin = lt;
        }
        string info()
        {
            cout << nama + " berwarna " + warna + " dan nama latinnya adalah " + latin + "\n\n" << endl;

            system("pause");
        }
};
```

Merupakan kelas anakan/kelas terderivasi/*subclass* pada program ini. Kelas inilah yang akan diwarisi oleh kelas induk sebelumnya. Dengan cara penulisan, setelah menulis nama *class* langsung diikuti oleh *public* dan nama kelas induknya. Begitu juga dengan dengan konstruktonya. Setelah menulis konstruktor, langsung diikuti oleh variabel yang digunakan termasuk yang ada dalam kelas induk. Untuk inisialisasi, hanya untuk variabel yang spesifik di kelas tersebut. Selanjutnya, membuat *method* untuk mengakses atribut yang ada. *Method* ini berguna untuk menampilkan hasil akhir beserta isi dari atribut nantinya.

```

int main()
{
    int pilih;

    do
    {
        system("cls");
        cout<<" Program Buah Jeruk"<<endl;
        cout<<" *****\n\n";
        cout<<" 1. Jeruk Manis\n";
        cout<<" 2. Jeruk Nipis\n";
        cout<<" 3. Jeruk Purut\n";
        cout<<" 4. Jeruk Keprok\n";
        cout<<" 5. Jeruk Mandarin\n";
        cout<<" 6. Keluar\n\n";
        cout<<" Pilih : "; cin>>pilih;
        switch(pilih)
        {
            case 1:
                {
                    jeruk manis("\n Jeruk Manis", "oranye sedang",
                        "Citrus x sinensis");
                    manis.info();

                    break;
                }
            case 2:
                {
                    jeruk nipis("\n Jeruk Nipis", "hijau", "Citrus
                        Aurantifolia");
                    nipis.info();
                }
        }
    }
}

```



```

        break;
    }
    case 3:
    {
        jeruk_purut("\n Jeruk Purut", "hijau", "Citrus x
hystric DC");
        purut.info();

        break;
    }
    case 4:
    {
        jeruk_keprok("\n Jeruk Keprok", "hijau kekuningan",
"Citrus Raticulata");
        keprok.info();

        break;
    }
    case 5:
    {
        jeruk_mandarin("\n Jeruk Mandarin", "oranye pekat",
"Citrus Reticulata");
        mandarin.info();

        break;
    }
    case 6:
    {
        cout<<"\n Bye Jeruk : ";
        break;
    }

```

```

        }
        default:
        {
            cout<<"\n Pilihaneun eopseo : ";
            system ("pause");
        }
    }
} while(pilih !=6);

system ("pause");
return 0;
}

```

Untuk program utama ini diawali dengan penulisan *int* diikuti dengan *main()* sebagai ciri dari program utama. Lalu, mendeklarasikan variabel pilih dengan tipe integer untuk memilih angka karena nantinya akan menggunakan menu. Masuk pada perulangan *do* untuk mengulangi proses sampai persyaratan terpenuhi pada bagian *while* pada akhir bagian *do* ini. Lalu pada bagian awal perintah *do* dibuat perintah *system("cls")* untuk membersihkan tampilan layar dari tampilan sebelumnya. Lalu dibuat lah tampilan menu sebagaimana mestinya. Diikuti oleh tempat memilih dimana inputan yang dimasukkan user akan membuat tampilan berubah dan bertambah sesuai inputan. Selanjutnya, ada bagian *switch(pilih)* yaitu untuk menentukan mana yang akan keluar jika inputan seperti ini atau seperti itu. Tentunya, setelah itu ada *case* dan nomor yang mengikutinya sebagai acuan jika kita memilih mana, apakah yang akan muncul. Pada setiap nomor ada aplikasi dari PBO yaitu memanggil objek dan memasukkan atribut didalamnya. Perintah *break* untuk menghentikan proses perulangan dan langsung masuk pada bagian dibawah blok perulangan. Di akhir *switch* ada perintah *default* yang berisi sesuatu Ketika kita menginput sesuatu yang tidak ada dalam menu. Perintah *system("pause")* untuk memberikan instruksi pada program untuk menghentikan proses eksekusi sampai user menekan sembarang tombol. Dampak penggunaan instruksi ini adalah munculnya teks "*Press any key to continue...*". Di

akhir blok perulangan ada bagian persyaratan yaitu `while()` yang berisi persyaratan yang harus terpenuhi. Jika tidak terpenuhi maka default yang akan muncul. *Return* 0 sebagai penanda bahwa program akan memiliki nilai balikan 0. Berikut adalah hasil output dari program ini.

```
Program Buah Jeruk
*****

1. Jeruk Manis
2. Jeruk Nipis
3. Jeruk Purut
4. Jeruk Keprok
5. Jeruk Mandarin
6. Keluar

Pilih : 1

Jeruk Manis berwarna oranye sedang dan nama latinnya adalah Citrus x sinensis

Press any key to continue . . .
-----
Process exited after 13.98 seconds with return value 3221225477
Press any key to continue . . .
```

Gambar 2.1.1

```
Pilih : 2

Jeruk Nipis berwarna hijau dan nama latinnya adalah Citrus Aurantifolia

Press any key to continue . . .
```

Gambar 2.1.2

```
Pilih : 3

Jeruk Purut berwarna hijau dan nama latinnya adalah Citrus x hystrie DC

Press any key to continue . . .
```

Gambar 2.1.3

```
Pilih : 4
Jeruk Keprok berwarna hijau kekuningan dan nama latinnya adalah Citrus Reticulata
Press any key to continue . . .
```

Gambar 2.1.4

```
Pilih : 5
Jeruk Mandarin berwarna oranye pekat dan nama latinnya adalah Citrus Reticulata
Press any key to continue . . .
```

Gambar 2.1.5

```
Pilih : 6
Bye Jeruk : Press any key to continue . . .
```

Gambar 2.1.6

2.2 Program Pewarisan Buah Apel

Program pewarisan kedua adalah program pewarisan dengan buah apel sebagai objeknya karena masih berkaitan dengan Pemrograman Berorientasi Objek (PBO).

```
#include <iostream>
#include <string>
```

Pada bagian awal, adalah dimana *library* yang diperlukan akan diberi perintah *include*. *Include* berarti memasukkan *library* di sebelahnya. *Iostream* merupakan *library* yang memiliki sintaks yang paling sering digunakan yaitu *cin* dan *cout*. *String* adalah *library* yang khusus menampung tipe data *string*. Tanpa memanggil *library string* maka tipe data *string* tidak dapat digunakan.

```
using namespace std;
```

Using *namespace std* untuk memudahkan pemanggilan kata perintah tanpa embel-embel *std::*.

```
class buah
{
    protected:
        string nama;
    public:
        buah(string nm): nama()
        {
            nama = nm;
        }
};
```

Merupakan kelas induk/kelas basis/*superclass* dalam program ini. Memiliki satu konstruktor pada bagian *public* dan satu variabel pada bagian *protected*. Variabel inilah yang nantinya akan diwariskan pada kelas anak.

```
class apel : public buah
{
    private:
        string warna;
        string latin;
    public:
        apel(string nm, string wrn, string lt ) : nama(nm), warna(wrn), latin(lt)
        {
            warna = wrn;
            latin = lt;
        }

        string info()
```

```

{
    cout << nama + " berwarna " + warna + " dan nama latinnya adalah " + latin
    + "\n\n" << endl;
    system("pause");
}
};

```

Merupakan kelas anakan/kelas terderivasi/*subclass* pada program ini. Kelas inilah yang akan diwarisi oleh kelas induk sebelumnya. Dengan cara penulisan, setelah menulis nama *class* langsung diikuti oleh *public* dan nama kelas induknya. Begitu juga dengan dengan konstruktonya. Setelah menulis konstruktor, langsung diikuti oleh variabel yang digunakan termasuk yang ada dalam kelas induk. Untuk inisialisasi, hanya untuk variabel yang spesifik di kelas tersebut. Selanjutnya, membuat *method* untuk mengakses atribut yang ada. *Method* ini berguna untuk menampilkan hasil akhir beserta isi dari atribut nantinya.

```

int main()
{
    int pilih;

    do
    {
        system("cls");
        cout<<" Program Buah Apel"<<endl;
        cout<<" -----\\n\\n";
        cout<<" 1. Apel Malang/Rome Beauty\\n";
        cout<<" 2. Apel Fuji\\n";
        cout<<" 3. Apel Opal\\n";
        cout<<" 4. Apel Merah\\n";
        cout<<" 5. Apel Hijau\\n";
        cout<<" 6. Keluar\\n\\n";
        cout<<" Pilih : "; cin>>pilih;
    }
}

```

```
switch(pilih)
{
    case 1:
        {
            apel malang("\n Apel Rome Beauty", "hijau dengan
            semburat merah", "Malus domestica 'Rome'");
            malang.info();

            break;
        }
    case 2:
        {
            apel fuji("\n Apel Fuji", "merah dengan semburat
            kuning", "Malus pumila");
            fuji.info();

            break;
        }
    case 3:
        {
            apel opal("\n Apel Opal", "kuning", "Malus
            domestica 'Opal'");
            opal.info();

            break;
        }
    case 4:
        {
            apel merah("\n Apel Merah", "merah", "Malus
            domestica 'Red Delicious'");
            merah.info();
        }
}
```

```

        break;
    }
    case 5:
    {
        apel hijau("\n Apel Hijau", "hijau", "Malus domestica");
        hijau.info();

        break;
    }
    case 6:
    {
        cout<<"\n Bye Apel : ";
        break;
    }
    default:
    {
        cout<<"\n Pilihaneun eopseo : ";
        system ("pause");
    }
}
} while(pilih !=6);

system ("pause");
return 0;
}

```

Untuk program utama ini diawali dengan penulisan *int* diikuti dengan *main()* sebagai ciri dari program utama. Lalu, mendeklarasikan variabel pilih dengan tipe integer untuk memilih angka karena nantinya akan menggunakan menu. Masuk pada perulangan *do* untuk mengulangi proses sampai persyaratan

terpenuhi pada bagian *while* pada akhir bagian *do* ini. Lalu pada bagian awal perintah *do* dibuat perintah *system("cls")* untuk membersihkan tampilan layar dari tampilan sebelumnya. Lalu dibuat lah tampilan menu sebagaimana mestinya. Diikuti oleh tempat memilih dimana inputan yang dimasukkan user akan membuat tampilan berubah dan bertambah sesuai inputan. Selanjutnya, ada bagian *switch(pilih)* yaitu untuk menentukan mana yang akan keluar jika inputan seperti ini atau seperti itu. Tentunya, setelah itu ada *case* dan nomor yang mengikutinya sebagai acuan jika kita memilih mana, apakah yang akan muncul. Pada setiap nomor ada aplikasi dari PBO yaitu memanggil objek dan memasukkan atribut didalamnya. Dengan cara menuliskan nama konstruktor disapi nama objeknya. Objek ini akan berisi atribut yang sudah inputan programmer. Dengan nama objek kita juga dapat memanggil method yaitu dengan nama objek.nama method. Perintah *break* untuk menghentikan proses perulangan dan langsung masuk pada bagian dibawah blok perulangan. Di akhir *switch* ada perintah *default* yang berisi sesuatu Ketika kita menginput sesuatu yang tidak ada dalam menu. Perintah *system("pause")* untuk memberikan instruksi pada program untuk menghentikan proses eksekusi sampai user menekan sembarang tombol. Dampak penggunaan instruksi ini adalah munculnya teks "*Press any key to continue...*". Di akhir blok perulangan ada bagian persyaratan yaitu *while()* yang berisi persyaratan yang harus terpenuhi. Jika tidak terpenuhi maka default yang akan muncul. *Return 0* sebagai penanda bahwa program akan memiliki nilai balikan 0. Berikut adalah hasil output dari program ini.

```
Program Buah Apel
-----

1. Apel Malang/Rome Beauty
2. Apel Fuji
3. Apel Opal
4. Apel Merah
5. Apel Hijau
6. Keluar

Pilih : 1

Apel Rome Beauty berwarna hijau dengan semburat merah dan nama latinnya adalah Malus domestica 'Rome'

Press any key to continue . . .

-----
Process exited after 2.984 seconds with return value 3221225477
Press any key to continue . . .
```

Gambar 2.2.1

```
Pilih : 2

Apel Fuji berwarna merah dengan semburat kuning dan nama latinnya adalah Malus pumila

Press any key to continue . . .
```

Gambar 2.2.2

```
Pilih : 3

Apel Opal berwarna kuning dan nama latinnya adalah Malus domestica 'Opal'

Press any key to continue . . .
```

Gambar 2.2.3

```
Pilih : 4

Apel Merah berwarna merah dan nama latinnya adalah Malus domestica 'Red Delicious'

Press any key to continue . . .
```

Gambar 2.2.4

```
Pilih : 5  
Apel Hijau berwarna hijau dan nama latinnya adalah Malus domestica  
Press any key to continue . . .
```

Gambar 2.2.5

```
Pilih : 6  
Bye Apel : Press any key to continue . . .
```

Gambar 2.2.6

BAB III

KESIMPULAN

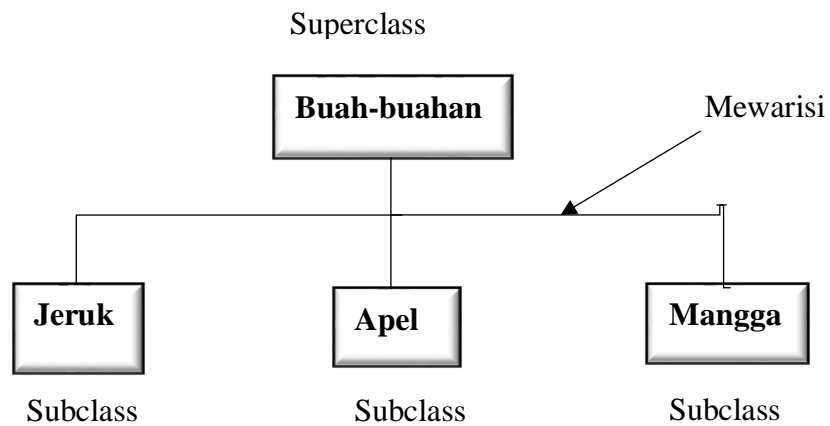
E1 enter

Pewarisan adalah sebuah tanda relas “is-a” antara kelas induk dan kelas anak. Cara ini memudahkan untuk membuat *class* yang memiliki kemiripan satu sama lain namun yang satu secara umum dan yang lain secara spesifik. Selain itu, pewarisan juga dapat diwariskan secara terus menerus asalkan ada sesuatu yang diwariskan pada kelas setelahnya. Anggapannya, kelas yang awalnya *subclass* berubah menjadi induk 2 dan memiliki *subclass* baru. Tentu tidak mengubah kenyataan bahwa induk 2 merupakan sebuah *subclass* karena juga masih memiliki induk di atasnya.

DAFTAR PUSTAKA

- Dosen Teknik Informatika. Algoritma dan Pemrograman. 2020. *Modul Praktikum Algoritma dan Pemrograman II*. Universitas Palangka Raya. Fakultas Teknik. Jurusan Teknik Informatika.
- Fajar. 2019. *Penjelasan dan Tipe-tipe Pewarisan (Inheritance)*.
<https://www.belajarcpp.com/tutorial/cpp/inheritance/> (Diakses pada Selasa, 14 April 2020 pada pukul 13.36 WIB)
- Santoso, Rachmat. 2017. *C++ - Inheritance (Dasar Pewarisan)*.
<http://www.nblognlife.com/2017/04/c-inheritance-dasar-pewarisan.html>
(Diakses pada Selasa, 14 April 2020 pada pukul 13.49 WIB)
- Andre. 2014. *Tutorial Belajar OOP PHP Part 9: Pengertian Inheritance (Pewarisan)*.
<https://www.duniaikom.com/tutorial-belajar-oop-php-pengertian-inheritance-pewarisan/> (Diakses pada Selasa, 14 April 2020 pada pukul 14.24 WIB)

LAMPIRAN



Gambar 1.2.1

```
Program Buah Jeruk
*****

1. Jeruk Manis
2. Jeruk Nipis
3. Jeruk Purut
4. Jeruk Keprok
5. Jeruk Mandarin
6. Keluar

Pilih : 1

Jeruk Manis berwarna oranye sedang dan nama latinnya adalah Citrus x sinensis

Press any key to continue . . .
-----
Process exited after 13.98 seconds with return value 3221225477
Press any key to continue . . .
```

Gambar 2.1.1

```
Pilih : 2

Jeruk Nipis berwarna hijau dan nama latinnya adalah Citrus Aurantifolia

Press any key to continue . . .
```

Gambar 2.1.2

```
Pilih : 3

Jeruk Purut berwarna hijau dan nama latinnya adalah Citrus x hystrie DC

Press any key to continue . . .
```

Gambar 2.1.3

```
Pilih : 4

Jeruk Keprok berwarna hijau kekuningan dan nama latinnya adalah Citrus Reticulata

Press any key to continue . . .
```

Gambar 2.1.4

```
Pilih : 5

Jeruk Mandarin berwarna oranye pekat dan nama latinnya adalah Citrus Reticulata

Press any key to continue . . .
```

Gambar 2.1.5

```
Pilih : 6

Bye Jeruk : Press any key to continue . . .
```

Gambar 2.1.6

```
Program Buah Apel
-----

1. Apel Malang/Rome Beauty
2. Apel Fuji
3. Apel Opal
4. Apel Merah
5. Apel Hijau
6. Keluar

Pilih : 1

Apel Rome Beauty berwarna hijau dengan semburat merah dan nama latinnya adalah Malus domestica 'Rome'

Press any key to continue . . .

-----
Process exited after 2.984 seconds with return value 3221225477
Press any key to continue . . .
```

Gambar 2.2.1

```
Pilih : 2  
Apel Fuji berwarna merah dengan semburat kuning dan nama latinnya adalah Malus pumila  
Press any key to continue . . .
```

Gambar 2.2.2

```
Pilih : 3  
Apel Opal berwarna kuning dan nama latinnya adalah Malus domestica 'Opal'  
Press any key to continue . . .
```

Gambar 2.2.3

```
Pilih : 4  
Apel Merah berwarna merah dan nama latinnya adalah Malus domestica 'Red Delicious'  
Press any key to continue . . .
```

Gambar 2.2.4

```
Pilih : 5  
Apel Hijau berwarna hijau dan nama latinnya adalah Malus domestica  
Press any key to continue . . .
```

Gambar 2.2.5

```
Pilih : 6  
Bye Apel : Press any key to continue . . .
```

Gambar 2.2.6