

**LAPORAN HASIL PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN II**



**NAMA : ERVINA AYU ROSAMADA**  
**NIM : 193020503020**  
**KELAS : A**  
**MODUL : III (POLIMORFISME)**

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

**2020**

**LAPORAN HASIL PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN II**



**NAMA : ERVINA AYU ROSAMADA**

**NIM : 193020503020**

**KELAS : A**

**MODUL : III (POLIMORFISME)**

Komposisi	MAX	Nilai
BAB I Tujuan dan Landasan Teori	10	9
BAB II Pembahasan	60	50
BAB III Kesimpulan	20	13
Daftar Pustaka	5	4
Lampiran	5	5
Jumlah	100	

**Penilai**  
**Asisten Praktikum**

**Diana**

## **BAB I**

### **TUJUAN DAN LANDASAN TEORI**

#### **I. TUJUAN**

Setelah menyelesaikan modul ini, mahasiswa diharapkan mampu membuat polimorfisme.

#### **II. DASAR TEORI**

Terdapat dua tipe polimorfisme, yaitu saat pengkompilasian (compile time) dan saat runtime. Polimorfisme saat pengkompilasian (compile time) dalam hal ini adalah overloading pada fungsi dan overloading pada operator yang pernah saya bahas di postingan sebelumnya (Baca C++ - Konsep Overloading Terhadap Fungsi dan C++ -Overloading terhadap Operator).

Pada polimorfisme saat runtime, fungsi pada kelas dasar di deklarasikan untuk umum, selanjutnya secara spesifiknya akan diterangkan kembali pada kelas turunan.

Analoginya, semua binatang memiliki suara maka pada kelas dasar diciptakan variabel dan fungsi untuk menampilkan suara. Namun seperti yang kita ketahui setiap jenis binatang memiliki suara yang berbeda, sehingga pada kelas turunan ditulis kembali suara yang khas dari jenis binatang tersebut.

Contoh lain, seorang raja (kelas dasar) mempunyai uang yang akan diberikan (diwariskan/diturunkan) ke anak kecil (kelas turunan) dan orang tua (kelas turunan). Uang beserta jumlahnya sama, namun perlakukannya dapat berbeda-beda (memiliki banyak bentuk). Oleh anak kecil uang tersebut digunakan untuk membeli mainan dan atau semua yang diinginkan. Oleh orang tua uang tersebut digunakan untuk membeli semua yang dibutuhkan termasuk kebutuhan anaknya.

Polymorphism adalah kemampuan untuk menggunakan operator atau fungsi dalam berbagai cara. Polimorfisme memberikan arti yang berbeda atau

funksinya kepada operator atau fungsi. Poly, merujuk ke banyak, menandakan banyak kegunaan dari operator dan fungsi ini. Fungsi tunggal penggunaan atau operator berfungsi dalam banyak cara bisa disebut polimorfisme. Polimorfisme mengacu pada kode, operasi atau benda yang berperilaku berbeda dalam konteks yang berbeda.

Polimorfisme adalah fitur yang kuat dari bahasa pemrograman berorientasi obyek C ++. Sebuah operator + berperilaku berbeda dalam konteks yang berbeda, seperti integer, float atau string yang mengacu konsep polimorfisme. Konsep di atas mengarah ke operator overloading. Konsep overloading juga merupakan cabang dari polimorfisme. Ketika keluar operator atau fungsi yang beroperasi pada tipe data baru itu kelebihan beban. Polimorfisme fitur ini mengarah pada konsep metode virtual.

Polimorfisme mengacu pada kemampuan untuk memanggil fungsi-fungsi yang berbeda dengan menggunakan hanya satu jenis fungsi panggil. Misalkan seorang programmer ingin kode kendaraan dari berbagai bentuk-bentuk seperti lingkaran, bujur sangkar, persegi panjang, dll Salah satu cara untuk mendefinisikan masing-masing dari kelas-kelas ini adalah memiliki fungsi anggota untuk masing-masing yang membuat kendaraan dari masing-masing bentuk. Pendekatan nyaman lain pemrogram dapat mengambil adalah untuk mendefinisikan sebuah basis kelas bernama Shape dan kemudian membuat sebuah instance dari kelas tersebut. Programmer dapat memiliki array yang menyimpan pointer ke semua objek yang berbeda dari kendaraan diikuti oleh struktur loop sederhana untuk membuat kendaraan, sesuai dengan bentuk yang dikehendaki, dengan memasukkan pointer ke array yang didefinisikan. Pendekatan ini mengarah kepada fungsi yang berbeda dieksekusi oleh panggilan fungsi yang sama. Polimorfisme digunakan untuk memberikan makna yang berbeda untuk konsep yang sama. Ini adalah dasar untuk pelaksanaan fungsi Virtual.

Dalam polimorfisme, satu fungsi atau operator berfungsi dalam banyak hal tergantung pada penggunaan berfungsi dengan benar. Agar hal ini terjadi, kondisi berikut harus berlaku:

1. Semua kelas-kelas yang berbeda harus berasal dari satu kelas dasar. Dalam contoh di atas, bentuk-bentuk kendaraan (lingkaran, segitiga, persegi panjang) yang dari satu kelas dasar yang disebut Shape.
2. Fungsi anggota harus dinyatakan virtual di kelas dasar. Dalam contoh di atas, fungsi anggota untuk membuat kendaraan harus dibuat sebagai virtual untuk kelas dasar.

Polimorfisme Memiliki arti “banyak bentuk”, melakukan hal yang sama untuk berbagai data yang berbeda, mengirimkan pesan yang sama ke berbagai objek yang berbeda karena tiap objek memberi respons dengan cara yang berbeda. Berikut ini merupakan contoh polimorfisme.

Polimorfisme memiliki syarat-syarat sebagai berikut: □ Ada hirarki pewarisan □

□ □ Kelas dalam hirarki pewarisan harus memiliki fungsi virtual (virtual method) dengan signature yang sama □ □ □ Menggunakan pointer atau rujukan ke kelas induk. Pointer digunakan untuk memanggil fungsi virtual □

Polimorfisme dapat diimplementasikan dengan menggunakan dasar function overriding (melakukan redefinisi suatu fungsi di kelas anak, fungsi yang di-override memiliki signature sama, signature sama : tipe balik, nama fungsi, parameter sama) dan pewarisan.

Suatu kelas disebut abstrak apabila memiliki minimal satu fungsi abstrak. Fungsi abstrak merupakan fungsi yang tidak memiliki definisi (hanya deklarasi fungsi)/menggunakan fungsi virtual (pure virtual). virtual balikan namaFungsi (parameter) = 0

## BAB II

### PEMBAHASAN

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

using namespace std;

class hewan {
    public:
        void data()
        {
            cout<<"DATA HEWAN"<<endl;
        }
};

class burung: public hewan{
    public:
        void data()
        {
            cout<<"info burung"<<endl;

            cout<<"_____
_____|"<<endl;

            cout<<"jumlah kakinya ada dua"<<endl;
            cout<<"memiliki sayap yang digunakan untuk
terbang"<<endl;
        }
};

class ikan: public hewan{
```

```

public:
    void data()
    {
        cout<<"info ikan"<<endl;

        cout<<"_____
_____|"<<endl;

        cout<<"ikan hidup di air"<<endl;
        cout<<"bernapas menggunakan insang"<<endl;
    }
};

int main(){
    hewan *h;
    burung *b;
    ikan *i;
    cout<<"=====
=====|"<<endl;
    h->data();
    cout<<"=====
=====|"<<endl;
    cout<<endl;
    b->data();
    cout<<"=====
=====|"<<endl;
    cout<<endl;
    i->data();
    cout<<"=====
=====|"<<endl;
    getch();
}

```

```
#include <iostream>
```

Tanda # disebut sebagai preprocessor directive, #include adalah sebuah preprocessor pengarah yang menunjukkan kepada compiler untuk meletakkan kode dari header file iostream.

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

Bagian ini berfungsi untuk mendeklarasikan beberapa fungsi pustaka yang dibutuhkan untuk antarmuka konsol, biasanya tersedia hanya di pustaka kompilator C untuk system DOS, Windows 3x, Phar Lap, IBM OS/2, or Win32, dan tidak tersedia di pustaka kompilator C untuk UNIX dan Linux.

```
using namespace std;
```

Menyatakan bahwa program akan menggunakan namespace yang bernama std, sehingga tidak perlu untuk menyertakan identitas std:: pada anggota dari namespace std salah satunya adalah cout.

```
class hewan {
```

Class adalah rancangan yang mendefinisikan variabel dan method-method pada seluruh objek tertentu. Class berfungsi untuk menampung isi dari program yang akan dijalankan suatu program. Class juga adalah sebuah konsep, konsep tersebut ada dua yaitu : method dan attribute. Tanda { berfungsi untuk memulai program.

```
public:
```

Public berfungsi untuk memberikan hak akses secara umum (public) kepada kelas-kelas turunannya maupun terhadap lingkungan luar didalam program. Bagian ini biasanya berisi fungsi-fungsi yang akan dijadikan sebagai penghubung (interface).

```
void data()
```

Fungsi pada tipe void tidak akan mengembalikan nilai apapun, void biasanya digunakan untuk menjalankan rutin program.



```
{
```

Berfungsi untuk memulai program.

```
cout<<"DATA HEWAN"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
}
```

Berfungsi untuk mengakhiri program.

```
};
```

Berfungsi untuk mengakhiri program.

```
class burung: public hewan{
```

Class adalah rancangan yang mendefinisikan variabel dan method-method pada seluruh objek tertentu. Class berfungsi untuk menampung isi dari program yang akan dijalankan suatu program. Class juga adalah sebuah konsep, konsep tersebut ada dua yaitu : method dan attribute. Public berfungsi untuk memberikan hak akses secara umum (public) kepada kelas-kelas turunannya maupun terhadap lingkungan luar didalam program. Bagian ini biasanya berisi fungsi-fungsi yang akan dijadikan sebagai penghubung (interface).

```
public:
```

Public berfungsi untuk memberikan hak akses secara umum (public) kepada kelas-kelas turunannya maupun terhadap lingkungan luar didalam program. Bagian ini biasanya berisi fungsi-fungsi yang akan dijadikan sebagai penghubung (interface).

```
Void data()
```

Fungsi pada tipe void tidak akan mengembalikan nilai apapun, void biasanya digunakan untuk menjalankan rutin program.

```
{
```

Berfungsi untuk memulai program.

```
cout<<"info burung"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
Cout<<"_____|"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
cout<<"jumlah kakinya ada dua"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada

layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
cout<<"memiliki sayap yang digunakan untuk terbang"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
}
```

Berfungsi untuk mengakhiri program.

```
};
```

Berfungsi untuk mengakhiri program.

```
class ikan: public hewan{
```

Class adalah rancangan yang mendefinisikan variabel dan method-method pada seluruh objek tertentu. Class berfungsi untuk menampung isi dari program yang akan dijalankan suatu program. Class juga adalah sebuah konsep, konsep tersebut ada dua yaitu : method dan attribute. Public berfungsi untuk memberikan hak akses secara umum (public) kepada kelas-kelas turunannya maupun terhadap lingkungan luar didalam program. Bagian ini biasanya berisi fungsi-fungsi yang akan dijadikan sebagai penghubung (interface).

```
public:
```

Public berfungsi untuk memberikan hak akses secara umum (public) kepada kelas-kelas turunannya maupun terhadap lingkungan luar didalam

program. Bagian ini biasanya berisi fungsi-fungsi yang akan dijadikan sebagai penghubung (interface).

```
void data()
```

Fungsi pada tipe void tidak akan mengembalikan nilai apapun, void biasanya digunakan untuk menjalankan rutin program.

```
{
```

Berfungsi untuk memulai program.

```
cout<<"info ikan"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
cout<<"_____|"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
cout<<"ikan hidup di air"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda <<

adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
cout<<"bernapas menggunakan insang"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
}
```

Berfungsi untuk mengakhiri program.

```
};
```

Berfungsi untuk mengakhiri program.

```
int main(){
```

Int main() merupakan fungsi (function) utama, fungsi yang akan dibaca pertama kali oleh kompilator secara otomatis, tidak akan function yang dibaca otomatis oleh kompilator kecuali function utama int main(), fungsi utama adalah program yang mengatur arah kompilator. Int dalam fungsi utama adalah sebuah return type integer. Tanda () merupakan tempat mengisi parameter untuk function, tetapi dalam fungsi utama, hal itu tidak dibutuhkan dan dibiarkan dalam keadaan kosong. Tanda { berfungsi untuk memulai program.

```
hewan *h;
```

Ptr yang dimaksud adalah pointer, pointer berfungsi sebagai penyimpan alamat memori dari sebuah variable. Selain itu pointer juga dapat digunakan untuk menyimpan alamat dari sebuah fungsi.

```
burung *b;
```

Ptr yang dimaksud adalah pointer, pointer berfungsi sebagai penyimpan alamat memori dari sebuah variable. Selain itu pointer juga dapat digunakan untuk menyimpan alamat dari sebuah fungsi.

```
ikan *i;
```

Ptr yang dimaksud adalah pointer, pointer berfungsi sebagai penyimpan alamat memori dari sebuah variable. Selain itu pointer juga dapat digunakan untuk menyimpan alamat dari sebuah fungsi.

```
cout<<"=====|"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
h->data();
```

Bagian program ini berfungsi untuk memanggil alamat yang berada pada pointer.

```
cout<<"=====|"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput.

Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
cout<<endl;
```

Bagian program ini berfungsi untuk ganti baris atau enter.

```
b->data();
```

Bagian program ini berfungsi untuk memanggil alamat yang berada pada pointer.

```
cout<<"=====|"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput. Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
cout<<endl;
```

Bagian program ini berfungsi untuk ganti baris atau enter.

```
i->data();
```

Bagian program ini berfungsi untuk memanggil alamat yang berada pada pointer.

```
cout<<"=====|"<<endl;
```

Cout adalah sebuah objek yang digunakan untuk menampilkan teks di layar monitor dan untuk mengarahkan data ke dalam standar output. Tanda << adalah sebagai operator pemasukan (insertion operators) dan berfungsi untuk menunjukkan pada compiler menampilkan output sesuai dengan yang diinput.

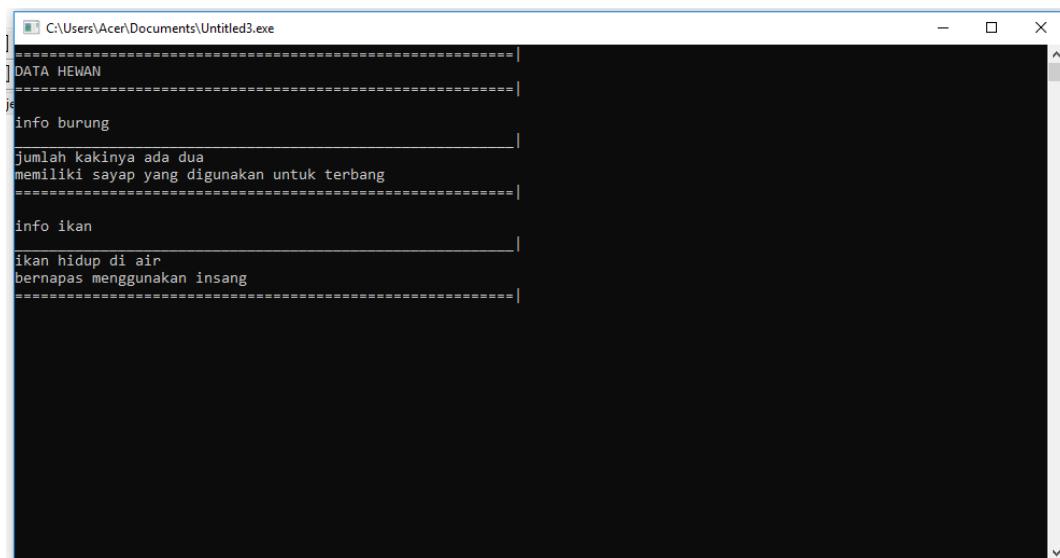
Tanda “ untuk menampilkan kata atau kalimat yang terdapat didalamnya kepada layar monitor. Endl merupakan anggota dari iostream dan masuk sebagai anggota namespace std yang digunakan untuk mengakhiri baris atau pindah baris (enter) pada hasil layar program.

```
getch();
```

Berfungsi untuk menahan proses yang sedang berjalan dan digunakan juga untuk membaca input dari keyboard.

```
}
```

Berfungsi untuk mengakhiri program.



```
1 =====|
2 DATA HEWAN|
3 =====|
4 je|
5 info burung|
6 =====|
7 jumlah kakinya ada dua|
8 memiliki sayap yang digunakan untuk terbang|
9 =====|
10 info ikan|
11 =====|
12 ikan hidup di air|
13 bernapas menggunakan insang|
14 =====|
```

*Gambar 1.1 tampilan output program*



## **BAB III**

### **KESIMPULAN**

Dalam OOP, konsep polimorfisme memungkinkan digunakannya suatu interface yang sama untuk memerintahkan suatu objek agar melakukan suatu aksi/tindakan yang mungkin secara prinsip sama tapi secara proses berbeda. Untuk mendeklarasikan sebuah fungsi virtual dalam kelas induk, cukup menambahkan katakunci virtual. Fungsi virtual murni dapat digunakan sebagai kelas abstrak. Kelas abstrak adalah kelas yang dideklarasikan tidak untuk menciptakan objek. Kelas abstrak mempunyai ciri paling tidak mengandung sebuah fungsi virtual murni.

## DAFTAR PUSTAKA

Rizky, Aditya.2011. *Tutorial Pemrograman Berorientasi Obyek dengan C++:Polimorfisme (Studi Kasus)*.

[https://adityarizky.net/tutorial-pemrograman-berorientasi-obyek-dengan-c-polimorfisme-studi-kasus/\(18/04/2020\)](https://adityarizky.net/tutorial-pemrograman-berorientasi-obyek-dengan-c-polimorfisme-studi-kasus/(18/04/2020))

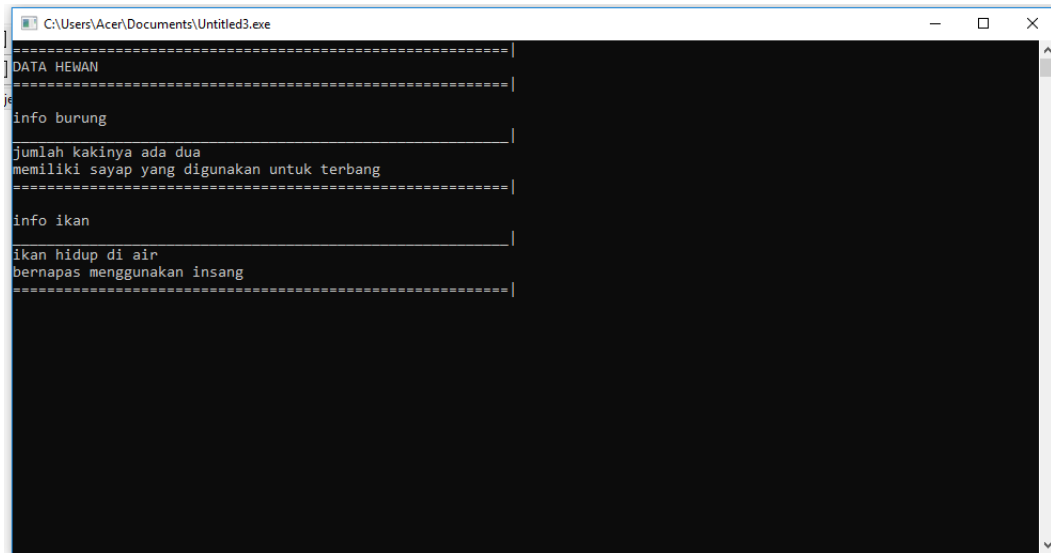
sinauarduino. 2016. *Polymorphism dan Virtual Function pada Classes C++*

[https://www.sinauarduino.com/artikel/polymorphism-dan-virtual-function-pada-classes-cpp/\(19/04/2020\)](https://www.sinauarduino.com/artikel/polymorphism-dan-virtual-function-pada-classes-cpp/(19/04/2020))

Kaulizt, Bill. 2013. *C++ POLIMORFISME DALAM PROGRAM C++*

[https://m.kaskus.co.id/thread/517cd0b433303000001/polimorfisme-dalam-program-c/\(19/04/2020\)](https://m.kaskus.co.id/thread/517cd0b433303000001/polimorfisme-dalam-program-c/(19/04/2020))

## LAMPIRAN



A screenshot of a Windows command prompt window titled "C:\Users\Acer\Documents\Untitled3.exe". The window has a black background with white text. The output of the program is as follows:

```
=====|
] DATA HEWAN=====|
js=====|
info burung=====|
jumlah kakinya ada dua=====|
memiliki sayap yang digunakan untuk terbang=====|
=====|
info ikan=====|
ikan hidup di air=====|
bernapas menggunakan insang=====|
=====|
```

*Gambar2.1 tampilan output program*