CAP444
OBJECT ORIENTED PROGRAMMING USING C++

Created By:
Kumar Vishal
(SCA), LPU

Overloading binary operators using friend function

Friend function takes two parameters in case when we want to overload binary operators using friend function

Ex:

friend A operator +(A &x, A &y);

Example:

What will be output for following code?

```
#include <iostream>
using namespace std;

class Sub
{
private:
   int a;
   int b;
public:
   Sub()
   {
      a=10;
      b=20;
   }
friend Sub operator -(Sub &x, Sub &y);
   void getresult()
   {
      cout<<a<<endl;
      cout<<b<<endl;
   }
};
```

```
Sub operator -(Sub &x,Sub &y)
{
      Sub z;
      z.a=x.b-x.a;
      z.b=y.b-y.a;
      return z;
}
int main()
{
  Sub a1,a2,a3;
  a3=a1-a2;
  a3.getresult();
  return 0;
}
```

A.
10
10

B.
-10
-10

C.
0
0

D. None

Situation??

Type Conversion

- Basic data types conversion done automatic by compiler
- User define data type conversion not done automatically
- User define data type conversion done by using either constructor or by using casting operator

What will be output?

```
#include <iostream>
 using namespace std;
 int main()
 {
    double a = 21.09399;
    float b = 10.20;
    int c ;
    c = a;
    cout << c ;
    c = b;
    cout << c ;
    return 0;
 }
```

A) 2110
B) 1210
C) 21
D) 121

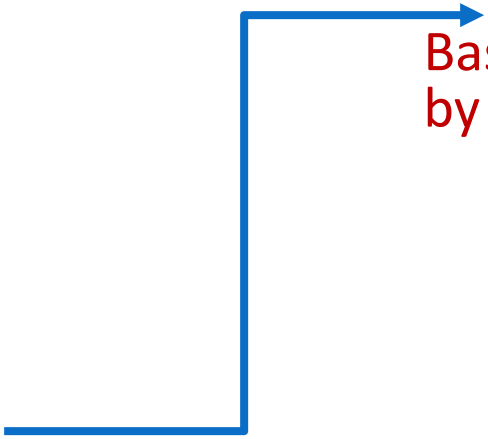
Three type of situation occurs during user define type conversion:

- 1. basic type to class type(using constructor)
- 2. class type to basic type(using casting operator function)
- 3. class type to class type (using constructor and casting operator function both)

basic type to class type(using constructor)

```
#include <iostream>
using namespace std;
class A
{

};
int main()
{
A a1;
int  x=8;
a1=x ;//basic to class type
   return 0;
}
```



Basic type to class type achieved by using constructor.

class type to basic type(using casting operator function)

Class type to basic type done by using casting operator function

1. It must be a define inside in class.
2. It must not specify a return type in function signature.
3. **It must not have any arguments.**

```
class A
{};
A a1;
int  x;
x=a1 //class type to basic type
```

Go through: cplusplus/Class to basic type conversionEx.pdf at master · vishalamc/cplusplus (github.com)

casting operator function

Syntax:

```
operator dest_typename()
{
  return type;
}
```

```
operator int()
{
  return a;
}
```

class type to class type (using constructor and casting operator function both)

Ex: A obj1; B obj2;

obj1 = obj2 ; // obj1 and obj2 are objects of different classes

- First approach using Constructor:-

Left side of assignment operator(=) which is class object we have to create constructor in that class here in Class A.

- Second approach using casting operator function:

Right side of assignment operator(=) which is class object we have to create casting operator function in that class here class B.

What will be out put for the following code?

```
 #include <iostream>
using namespace std;
class Circle
{
   int radius;
    Circle(){}

   Circle(int radius)
   {
    this->radius=radius;
    cout<<this->radius;
   }
};
```

```
int main()
{
   Circle a1, b1(5);
   Circle b2 = Circle(8);
   return 0;
}
```

A. 55
B. 88
C. 58
D. Error

What will be out put for the following code?

```
 #include <iostream>
using namespace std;
class Circle
{
   int radius;
public:
    Circle(){}

   Circle(int radius)
   {
    this->radius=radius;
    cout<<this->radius;
   }
};
```

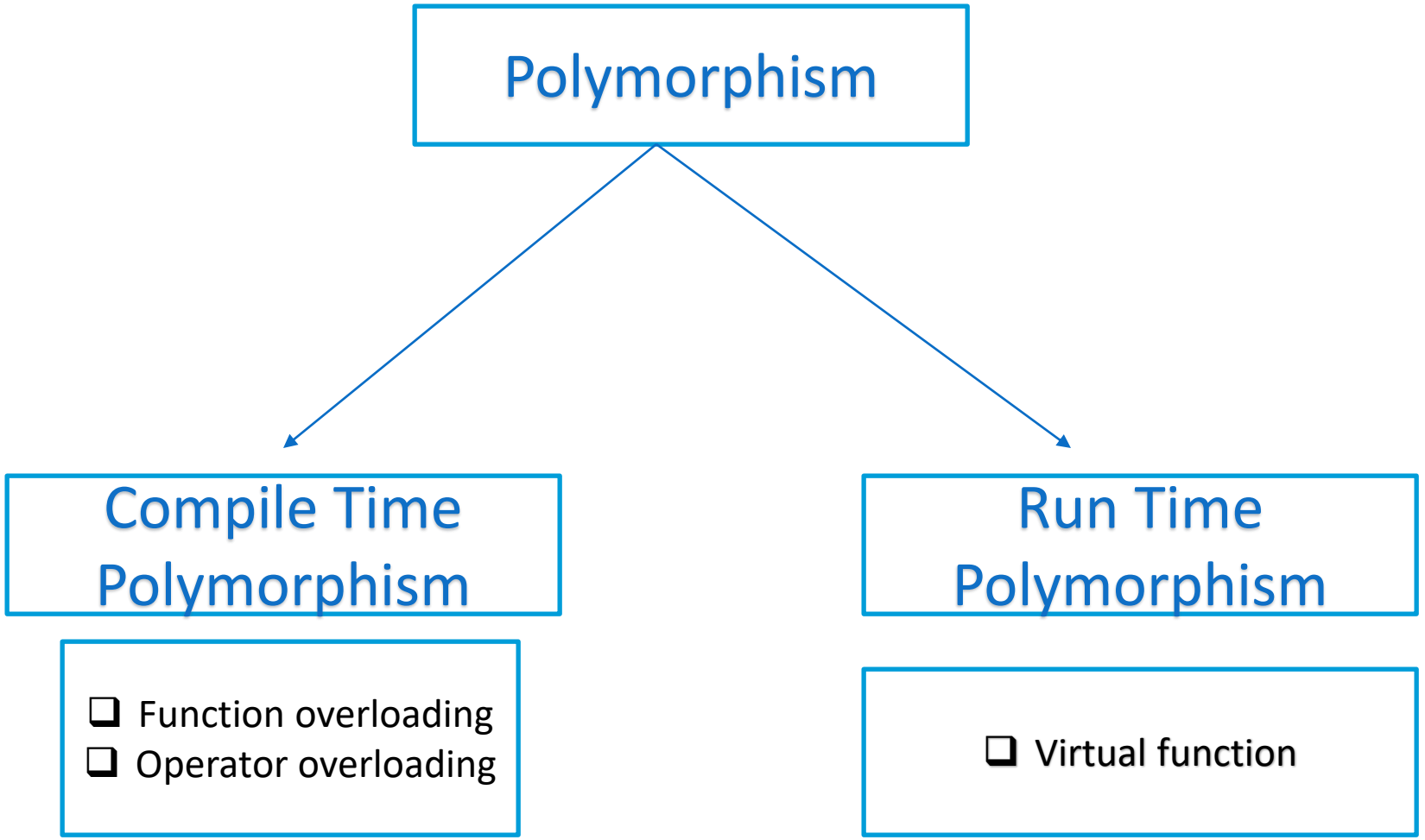
```
int main()
{
   Circle a1, b1(5);
   Circle b2 = Circle(8);
   return 0;
}
```

A. 55
B. 88
C. 58
D. Error

Unit-3

Run-time polymorphism and virtual functions :

- virtual base classes,
- abstract classes,
- pointer to object,
- this pointer,
- pointer to derived class,
- virtual function,
- pure virtual function,
- early vs late binding


Polymorphism
Compile Time Polymorphism
Run Time Polymorphism
❑ Function overloading
❑ Operator overloading
❑ Virtual function

Real Life Example: polymorphism (Run Time)

Preparing food according to hotel menu

Compile time)

During competition preparing food

(Run-time)


WINAMP
00:05
Fear (3:12) *** 2. At the Gates - E
192 kbps 44 kHz
mono stereo
EQ PL
SHUFFLE
WINAMP EQUALIZER
ON AUTO
PRESETS
+12 db
+0 db
-12 db
PREAMP 60 170 310 600 1K 3K 6K 12K 14K 16K
WINAMP PLAYLIST
1. DJ Mike Llama - Llama Whippin' Intro 0:05
2. At the Gates - Blinded by Fear 3:12
3. At the Gates - Slaughter Of The Soul 3:02
4. At the Gates - Cold 3:28
ADD REM SEL MISC
3:12/9:47+
00:05
LIST OPTS

Shuffle is On

Student

regNo, Name
studentDetails()

regNo, Name
studentDetails()

Exam

Project

regNo, Name
studentDetails()

regNo, Name
studentDetails()

Result ambiguity arises as to which data/function member would be called?

virtual base class introduce

virtual base classes

- It means we are making base class as virtual
- But why ???
- In situation???

Let's assume

virtual base class: Example


LOVELY PROFESSIONAL UNIVERSITY
PUNJAB (INDIA)
L
P
U

Any Query?