C++ BIG THREE

Problem Solving with Computers-II

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

int main(){
    cout<<"Hola Facebook\n";
    return 0;
}
```

Read the syllabus. Know what's required. Know how to get help.

Today’s learning goals:

We want to understand the what, why, and how of the Big Three:

- Destructor
- Copy constructor
- Copy assignment operator

Constructor and Destructor

Every class has the following special methods:

- Constructor: Called right AFTER an object is created in memory
- Destructor: Called right BEFORE an object is deleted from memory

The compiler automatically generates default versions, but you can provide user-defined implementations

```
void foo(){
    Complex p(1, 2);
    Complex *q = new Complex(3, 4);
}
```

What is the output?

A. 1 + 2j

B. 3 + 4j

C. 1 + 2j
   3 + 4j

D. None of the above

```
class Complex
{
private:
    double real;
    double imag;
public:
    Complex(double re = 0, double im = 0);
    ~Complex(){ print();}
    double getMagnitude() const;
    double getReal() const;
    double getImaginary() const;
    void print() const;
    void conjugate();
    void setReal(double r);
    void setImag(double r);
};
```

Copy constructor

- Parameterized constructor whose first argument is a class object
- Default behavior: **initializes a (new) object using an existing object**

In which of the following cases is the copy constructor called?

```
A.  Complex p1;
    Complex p2(1, 2);

B.  Complex p1(1, 2);
    Complex p2(p1);

C. Complex *p1 = new Complex(1, 2);
   Complex p2 = *p1;

D. B&C
E. A, B & C
```

Copy assignment (operator=)

- For existing objects x, y, this statement calls the operator= function:

  x = y;

- Default behavior: Copies the member variables of rhs object (y) to lhs object (x)

```
Complex x(1, 2);
Complex y;
y = x;
cout << y;
```

```
double foo(Complex p){
   return p.magnitude();
}
int main(){
    Complex q(1, 2);
    foo(q);
}
```

Which special method(s) is/are called as a result of calling `foo()`?

A. Parameterized constructor

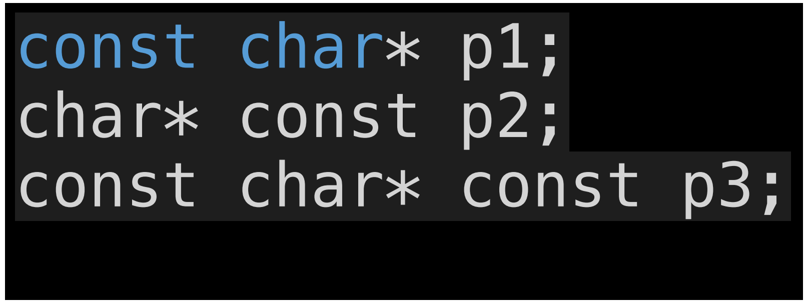
B. Destructor

C. Copy constructor

D. Copy assignment

E. All of the above

Constant pointers and pointers to constants



```
const char* p1;
char* const p2;
const char* const p3;
```

Summary

- Classes have member variables and member functions (method). An object is a variable where the data type is a class.
- You should know how to declare a new class type, how to implement its member functions, how to use the class type.
- Frequently, the member functions of an class type place information in the member variables, or use information that's already in the member variables.
- New functionality may be added using non-member functions, friend functions, and operator overloading
- If a class allocates data on the heap, then a user-defined destructor must be implemented to perform a clean-up procedure (de-allocate heap memory)

Next time

- Linked Lists and the rule of three