计算机学院面向对象程序设计终考试卷（样本）

班级_____ 学号11202082x姓名郑子申 成绩_____

Part I. Answer each questions by choosing A, B, C or D.

1. ___B___ is the default access permission for members in a class.
   a) public
   b) private
   c) protected
   d) static

2. If we have this code:
   class A { public: int a; };
   A *obj;
   ___C___ accesses the "a" variable.
   a) obj.a
   b) obj(a)
   c) obj->a
   d) obj::a

3. ___D___ identifies a constant pointer to changeable character data.
   a) const char&
   b) char& const
   c) const char* (指针本身可变，指针指向的内存不可变)
   d) char* const

4. ___B___ is the specific meaning of the word "reference" in C++ programming.
   a) Referring to the compiler's user manual.
   b) Referring to the same memory location.
   c) A document written by a referee
   d) Citing of a previous piece of research.

5. It's right that ___D___.
   a) A constructor must be called explicitly.
   b) A constructor can be a virtual function. (构造函数不能被定义为虚函数)
   c) A constructor cannot have default arguments.
   d) A constructor can call other member functions of its own class.

Part II. Fill in the blanks.

6. Write the output of the programs

```
#include <iostream.h>
class blah
{
public:
    blah(int x)   { a++; }
    static int a;
};
int blah::a = 0;
void main()
{
   blah b1(5);
   blah b2(12);
   cout << "b1.a = " << b1.a;
   cout << "b2.a = " << b2.a;
}
```

The output is:

_____

_____

_____

b1. a=2 b2.a=2

PartIII. Programming

7. According the definition of **Complex** class and main function, please finish the member functions of **Complex** class.

```
#include <iostream.h>
class Complex
{
private:
    float a, b;              // a is real part of a complex, b is image part of a complex
 public:
     // Finish the member functions according to main

};
void main()
{
    Complex c1(10, 20), c2(30, 40), c3;
    c3 = c1 + c2;
}
```

```
class Complex
{
 private:
     float a, b;
 public:
     Complex(float a0, float b0): a(a0), b(b0) {}
     complex () { a=b=0; }
     Complex& operator = ( Complex& c) {
           a = c.a;
           b= c.b;
           return *this;
      }
      Complex operator + ( Complex& c) {
           Complex ret;
           ret.a=  a + c.a;
           ret.b=  b+ c.b;
           return ret;
      }
 };
```