1. 选择：

有以下程序：

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

class A
{
public:
   A() { cout << 'A'; }
   ~A() { cout << 'C'; }
};

class B : public A
{
public:
   B() { cout << 'G'; }
   ~B() { cout << 'T'; }
};

int main()
{
   B obj;
   return 0;
}
```

执行后的输出结果是（ ）

A. GATC　　B. AGTC　　C. GACT　　D. AGCT

2．判断：

在下面的类中，**bar** 将先于 **foo** 被构造。（　）

```
#include "Mucus.h"
class Marvin {
   public:
      Marvin() : bar(2), foo(3) {}
      Mucus foo;
      Mucus bar;
};
```

3．改错：

```
class Time
{
public:
   void ~Time(int);
   int Time(int,int,int);
private:
   int hour = 0;
   int minute = 0;
   int second = 0;
};
```

4. 阅读代码：

```
#include <iostream>
using namespace std;
class blah {
  public:
      static int a;
      int b;
      blah(int x) {
         b=x;
         a=b+1;
      }
};

int blah::a = 0;

int main(void) {
  blah b1(5);
  blah b2(12);
  cout << "b1.a is " << b1.a << endl;
  cout << "b1.b is " << b1.b << endl;
  cout << "b2.a is " << b2.a << endl;
  cout << "b2.b is " << b2.b << endl;
  return 0;
}
```

5．问答题：

已知 **Base** 类和 **Derive** 类有如下继承关系：

```
class Base
{
  public:
      Base();
      virtual ~BaseV();
};
class Derived : public Base
{
  public:
      Derived();
      ~Derived();
      void PrintMessage();
};
```

请问：下面的函数是否有 **Bug**，如果有，请说明并修正。

```
void foo( Base* pb ) {
  Derived* pd = dynamic_cast<Derived*>( pb );
  pd->PrintMessage();
 };
```

6．编程题：

```
 class String
{
  public:

     String(const char *str = NULL);
     String(const String &other);
     ~String(void);
     String & operate =(const String &other);

  private:

     char *m_data;
};

//-------------------------------------------------
```

编写 **String** 的构造函数、析构函数、拷贝构造函数和赋值函数。