

东南大学 考试卷 (卷)

课程名称 程序设计基础及语言 B 考试学期 13-14-3

适用专业	计算机科学与技术	考试形式	半开卷	考试时间长度	笔试(60 分) 120 分钟 机试(40 分) 100 分钟 <hr/> 总分: 100 分
------	----------	------	-----	--------	--

试题	得分	评阅人
第一部分：笔试		
第二部分：机试		
成绩合计：		

第一部分 笔试

I. Read the following programs: (20 scores)

1. What does the following program output? (5 scores)

```
class MyClass
{
public:
    MyClass()
    {
        cnt++;
    }
    ~MyClass()
    {
        cnt--;
    }
    static int Count()
    {
        return cnt;
    }
private:
    static int cnt;
};

int MyClass::cnt = 0;

int main()
{
    cout << MyClass::Count() << ' ';
    MyClass t1, t2;
    MyClass* p1 = new MyClass;
    MyClass* p2 = new MyClass;
    cout << MyClass::Count() << ' ';
    delete p1;
    delete p2;
    cout << MyClass::Count() << endl;
    return 0;
}
```

2. What does the following program output? (5 scores)

```
class MyClass1
{
public:
    MyClass1(int i)
    {   x=i; };
    virtual void print()
    {   cout<<x<<endl; };
private:
    int x;
};
class MyClass2: public MyClass1
{
public:
    MyClass2(int i):MyClass1(i+10)
    {   x=i; };
    virtual void print()
    {
        MyClass1::print();
        cout<<x<<endl;
    };
private:
    int x;
};
int main()
{
    MyClass1 a(1);
    MyClass2 b(2);
    MyClass1 *s[2];
    s[0]=&a;
    s[1]=&b;
    for(int i=0;i<2;i++)
        s[i]->print();
    return 0;
}
```

3. What does the following program output? (5 scores)

```
class A{
public:
    A(int a){ cout << "A ctor!" << endl; }
    A(){ cout << "A default ctor!"<< endl; }
    ~A() { cout << "A dtor!" << endl; }
};
class B{
public:
    B(int a, int b){ cout<<"B ctor!"<<endl; }
    B(): m(0) { cout<<"B default ctor!"<<endl;}
    ~B(){ cout<<"B dtor!"<<endl;}
private:
    A m;
};
class C: public B{
public:
    C(int a, int b) { cout<<"C ctor!"<<endl; }
    ~C(){ cout<<"C dtor!"<<endl;}
private:
    A m;
};
int main()
{
    C obj(0, 0);
    return 0;
}
```

4. What does the following program output? (5 scores)

```
class Obj{
    int id;
public:
    Obj(int n){ id = n; cout << "ctor " << id << endl;}
    ~Obj(){ cout << "dtor " << id << endl;}
};
void f2( )
{
    double a = 0;
    try{
```

```

        throw a;
    }
    catch(...){
        cout<<"OK2!"<<endl;
        throw;
    }
    cout<<"end2"<<endl;
}
void f1( )
{
    try{
        f2( );
    }
    catch(char){
        cout<<"OK1!"<<endl;
    }
    cout<<"end1"<<endl;
}
int main( )
{
    Obj o(0);
    try{
        Obj o(1);
        f1( );
    }
    catch(double){
        cout <<"OK0!"<<endl;
    }
    cout<<"end0"<<endl;

    return 0;
}

```

II. Fill in the blanks in each of the following to complete the programs.

(20 score)

1. (10 scores) According to the output, complete the following program:

```
template<typename T>
```

```

class Set
{
public:
    Set(int =10);
    ~Set()
    {   delete [] setPtr; }
    bool put(int index, T);
    T get(int index);
private:
    int size;
    T *setPtr;
};

template <typename T>
Set<T>::Set(int s)
{
    size=s;
    _____ (1)
}

template <typename T>
_____ (2)
{
    if(index>=0 && index<size)
    {
        _____ (3)
        return true;
    }
    else
        return false;
}

template <typename T>
T Set<T>::get(int index)
{
    if(index>=0 && index<size)
        _____ (4)
    else
        cout<< "Out of Range!"<<endl;
}

```

```

int main()
{
    Set<int> s1;
    for(int i=0;i<10;i++)
        s1.put(i,i+1);
    for(int i=0;i<10;i++)
        cout<<s1.get(i)<<" ";
    cout<<endl;
    _____
    (5)
    for(int i=0;i<5;i++)
        s2.put(i,(i+1)*1.1);
    for(int i=0;i<5;i++)
        cout<<s2.get(i)<<" ";
    cout<<endl;
    return 0;
}

```

Output:

```

1  2  3  4  5  6  7  8  9  10
1.1 2.2 3.3 4.4 5.5

```

2. Complete the following program, which writes the object of GradeBook class to the file “Gradebook.dat”, and then reads data from the file and prints to the screen.

```

#include <iostream>
#include <string>
_____
(1)
using namespace std;

class GradeBook{
    friend _____(2)_____ operator<<(ostream& output, const GradeBook &book){
        output << book.courseName << endl << book.teacherName << endl;

        _____(3)_____
    }
    friend istream& operator>>(istream& input, GradeBook &book){
        _____(4)_____
        return input;
    }
}

```

```

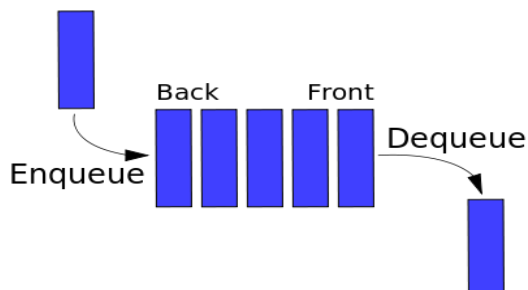
    }
public:
    GradeBook(string course, string teacher){
        courseName = course;
        teacherName = teacher;
    }
private:
    string courseName, teacherName;
};

int main()
{
    GradeBook book("C++", "TeacherName");
    ofstream fout("Gradebook.dat");
    fout << book;
    fout.close();
    _____(5)_____ // open the file Gradebook.dat
    fin >> book;
    fin.close();
    cout << book << endl;
    return 0;
}

```

III. Write programs according to the requests (20 score)

1. (10 scores) Define a class for polynomial $a*x^2+bx+c$, overload the +, - operator, and implement two functions, Compute to get its result, Root to get its root.
2. (10 scores) Compared with Stack, Queue is a First-In-First-Out (FIFO) data structure. As shown in the below figure, in a FIFO data structure, the first element added to the queue will be the first one to be removed. Write a program to implement and test Queue class-template, which should provides dequeue, enqueue, isEmpty, isFull member functions.



答案:

I-1:

0 4 2

I-2:

1

12

2

I-3:

ANS:

A ctor!

B default ctor!

A default ctor!

C ctor!

C dtor!

A dtor!

B dtor!

A dtor!

I-4:

ANS:

ctor 0

ctor 1

OK2!

dtor 1

OK0!

end0

dtor 0

II-1:

(1) setPtr=new T[size];

(2) bool Set<T>::put(int index, T value)
(3) *(setPtr+index)=value;
(4) return *(setPtr+index);
(5) Set<double> s2(5);

2.

```
#include <fstream>          ①  
ostream&                    ②  
return output;              ③  
input >> book.courseName >> book.teacherName; ④  
ifstream fin("Gradebook.dat"); ⑤
```

