东 南 大 学 考 试 卷（ A 卷）

课程名称	程序设计基础及语言 II（双语）	考试学期	19-20-3	卷面	60 分
适用专业	计算机大类	考试形式	半开卷 可带一本教材	考试时间长度	机试 120 分钟

仅 允 许 携 带 课 程 指 定 教 材

第二部分 机试

要求：

a) 在本地 D：或 E：盘中，建立自己的文件夹，用来完成程序的编写和调试。

b) 建议：第一题 Project(项目名)为 Pro1，第二题 Project(项目名)为 Pro2，以此类推。

III. To write and test programs according to the requirements（60 scores）

1. Complete the definition of class Log to make main function correct output. Note: to use dynamic memory allocation technique. The C++ standard libraries for string are not allowed to use. (20 scores)

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

class Log
{
public:
   Log(int , char *);
   int SearchSubStr( char * str);
   ~Log( );
     …
private:
   int id;
   char * info;
   int length;
};
```

```
void main()
{
      Log log1(1, "Every thing is OK!");
      Log log2(2, "It has a problem!");
      Log log3(3, "Every thing is OK!");
      if (log1 == log3)
            cout << "log1 == " << "log3" << endl;
      if( log1 != log2)
            cout << "log1 != " << "log2" << endl;

      cout << log1;

      int index = log1.SearchSubStr("thing");
      if (index > 0)
        cout << " Find the sub string, and the index is " <<
index<<endl;
      else
        cout << "Do not find the sub string.\n";
}
```

Output:

log1 == log3

log1 != log2

id is 1, and info is ' Every thing is OK! '

Find the sub string, and the index is 7

2. Please complete the implementation of shopping list class to make main function correct output.

Note: Refer to the following code and Don't add or delete any data member in the classes.

```
#include <iostream>
using namespace std;
class Book {
private:
    int id;
    double price;
public:
    Book(int i = 0, double p = 0){
        id = i;
        price = p;
    }
};
class Date {
private:
    int year, month, day;
public:
    Date() {
    cout << "Default constructor of Date\n";
    year = 2020; month = 8; day = 24;
    }
    Date(int y, int m, int d)
    {
        cout << "Constructor of Date\n";
        year = y; month = m; day = d;
    }
    ~Date()
    {
        cout << "Destructor of Date\n";
    }
};
```

```
class BookList {
private:
    Date date;
    int num;
    Book *list;
public:
    BookList() { num = 0; list = 0; }
    BookList(Book *p, int n, Date d){
        …..
    }
BookList(const BookList & s):date(s.date){
    …….
}
    void addBook(Book t)
    {
        ……
    }
    ~BookList()
    {
       cout << "Destructor of BookList\n";
       delete[] list;
    }
};
void main(){
    Book b1(1, 10), b2(2, 30);
    BookList booklist_1;
    booklist_1.addBook(b1);
    booklist_1.addBook(b2);
    BookList booklist_2 = booklist_1;
    cout<< booklist_2;
}
```

Output：

Default constructor of Date

Copy constructor

Year2020 month8 day24

Id1 price10

Id2 price30
Destructor of BookList
Destructor of Date
Destructor of BookList
Destructor of Date

3. Create a class called Rectangle (矩形) that includes two pieces of information as data members- width (type int) and height (type int). Your class should have a constructor with two parameters to initialize the two data members. Provide set and get functions for every data member, area and perimeter (周长)functions that calculates the area and perimeter of the rectangle. Provide a member function named draw to draw the rectangle's shape (形状) with '*'. Create a derived class FilledRectangle of Rectangle, it includes an extra date member- filledChar (type char) and necessary functions. The main function is given to test your class.

```
#include <vector>
int main() {
    Rectangle a(6, 5);
    FilledRectangle b(5, 7, '&');
    FilledRectangle c(7, 5, '#');
    vector<Rectangle*> rectangles;
    rectangles.push_back(&a);
    rectangles.push_back(&b);
    rectangles.push_back(&c);
    for (int i=0;i<3;++i) {
        cout<<"Rectangle "<<i+1<<endl;
        cout<<"Width "<<rectangles[i]->getW()
            <<" Height "<<rectangles[i]->getH()
            <<endl;
        cout<<"Area "<<rectangles[i]->area()
            <<" Perimeter "<<rectangles[i]->perimeter()
            <<endl;
        rectangles[i]->draw();
    }
}
```

The output should be:

```
Rectangle 1
Width 6 Height 5
Area 30 Perimeter 22
******
*    *
*    *
*    *
******
Rectangle 2
Width 5 Height 7
Area 35 Perimeter 24
*****
*&&&*
*&&&*
*&&&*
*&&&*
*&&&*
*****
Rectangle 3
Width 7 Height 5
Area 35 Perimeter 24
*******
*#####*
*#####*
*#####*
*******
```