

中山大学计算机科学系 2007 级
计算机科学与技术专业、网络工程专业、信息安全专业（ABCDE 班）
程序设计 B 卷

学号 _____ 姓名 _____ 成绩 _____

(试卷共 6 页，答案全写在答题纸上，交卷时连试卷一同交回)

考试形式：闭卷

任课老师：林瑛、肖菁、杨永红

2008-6



《中山大学授予学士学位工作细则》第六条：“考试作弊不授予学士学位。”

一、单项选择 (每小题 1 分，共 15 分)

1. 函数的类型不能是：
A) 基本数据类型 B) 数组类型 C) 结构类型 D) 指针类型
2. C++源程序文件经过编译后生成的目标程序的缺省扩展名为：
A) cpp B) exe C) obj D) lik
3. 派生类的成员函数可以访问：
A) 公有继承的基类公有成员 B) 公有继承的基类私有成员
C) 公有继承的基类保护成员 D) 私有继承的基类公有成员
4. 关于虚函数的描述中，正确的是：
A) 虚函数是一个 static 类型的成员函数
B) 虚函数是一个非成员函数
C) 基类中说明了虚函数后，派生类中对同名函数的重定义时可不说明为 virtual，其虚特性保持不变
D) 派生类的虚函数与基类的同名虚函数具有不同的参数个数和类型
5. 关于虚基类的描述，正确的是：
A) 虚基类的唯一副本只被初始化一次
B) 无论是虚基类还是普通基类，其构造函数的调用的次序取决于基类在声明时的次序
C) 类中对象成员的初始化先于虚基类副本的初始化
D) 虚基类的析构函数最先调用
6. 设运行环境中 int 类型长度为 2 字节，且 ptr 是一个 int 类型的指针，且 ptr 的值是 2400，ptr 指向的整数变量的值是 24，那么 *ptr+1 的值是：
A) 2401 B) 2402 C) 25 D) 26
7. 作为类型构造机制，如果类 DERIVE 以私有方式继承类 BASE，那么基类型为 BASE 的指针_____指向基类型为 DERIVE 的对象。
A) 一定能 B) 一定不能 C) 不一定能 D) 应该能
8. 若在一个类中用成员函数重载了某种二元运算符@，而 obj1 和 obj2 都是该类的对象，则表达式 obj1@obj2 被 C++编译器解释为：
A) obj1.operator@(obj2) B) obj2.operator@(obj1)
C) operator@(obj1,obj2) D) operator@(obj2,obj1)
9. 在 C++语言中，以下哪个表达式采用了十六进制表示整型常量：
A) k = 0123; B) k = '\123'; C) k = 123; D) k = 0x123;
10. 对以下声明的类 A，sizeof(A) 的值为：

```
class A {
    float std;
    static float max, min;
};
```

A) 4 B) 8 C) 12 D) 16

11. 下列将类 A 说明为类 B 的虚基类的语句中，正确的是：

A) class B: private A virtual B) class B: private virtual A
C) class B: virtual private A D) virtual class B: private A

12. 编译程序在对函数调用进行静态绑定时，根据以下因素决定调用哪一个函数，**不正确**的是：

A) 函数类型 B) 函数名 C) 实际参数个数 D) 实参相应位置的类型

13. 不能重载的运算符是：

A) :: [] ?: B) :: ?: # C) () ?: # D) . :: ->

14. 关于构造函数**不正确**的说法是：

A) 构造函数可以有返回值 B) 一个类可以有多个构造函数
C) 构造函数名与类名相同 D) 构造函数用于初始化新创建的类对象

15. template<class T>

```
class APPLE{.....};
```

定义类模板 APPLE 的成员函数的正确格式是：

A) T APPLE<T>::Push(T obj) {.....} B) T APPLE::Push(T obj) {.....}
C) template<class T> D) template<class T>
T APPLE::Push(T obj) {.....} T APPLE<T>::Push(T obj) {.....}

二、程序改错 (共 20 分)

1. 下列的程序片段是否语法或语义错误，如有请说明错误原因，并改正：

1) char look_up(char word, char dictionary, char word);

```
2) class LINE {
    public:
        int draw();
        int is_on_line(int x, int y);
    private:
        int start_x = 0, start_y = 0;
        int end_x, end_y;
};
```

```
3) int* age_ptr;
    int age = 30;
    *age_ptr = age;
```

```
4) template <class ITEM, class LINK>
    int compare(ITEM source, ITEM target)
    {    if (source > target) return 1;
        else return 0;
    }
```

2. 阅读下面的程序，指出在哪些行上有语法错误（共 3 个错误），说明错误原因，并改正：

```
1) #include <iostream.h>
2) #include <string.h>
3) class PERSON {
```

```

4)     public:
5)         PERSON(char* name)
6)         { int len;
7)           len = strlen(name);
8)           PERSON::name = new char[len + 1];
9)           strcpy(PERSON::name, name);
10)        }
11)        ~PERSON(int t) { delete []name; }
12)        char* get_name() { return name; }
13)    protected:
14)        char* name;
15) };
16) class STUDENT: PERSON {
17)     public:
18)         void STUDENT(char* stname, int scr): score(scr), PERSON(st_name)
19)         { int len;
20)           len = strlen(stname);
21)           name = new char[len + 1];
22)           strcpy(name, stname);
23)         }
24)         int get_score() { return score; }
25)     protected:
26)         int score = 100;
27) };
28) void main()
29) { STUDENT soft("Zhao", 80);
30)   cout << "Student: " << soft.get_name();
31) }

```

三、程序输出 (共 20 分)

1. 下面程序根据录入的年、月、日求该天是星期几，其中的公式来自蔡勒(Zeller)。假如程序输入的三个数分别为 1971、2、26，即其中的 year, month, day 分别是 1971、2 和 26，试给出程序的输出结果。

```

#include <iostream.h>
void main()
{   int year, month, day, centry;
    int temp_year, temp_month;
    int week_day;

    cout << "\nPlease input the year, month and day: ";
    cin >> year >> month >> day;

    temp_year = year; temp_month = month;
    if (month == 1 || month == 2) {
        temp_year = temp_year - 1;
        temp_month = temp_month + 12;
    }

    centry = temp_year / 100; temp_year = temp_year % 100;
    if (centry == 0) centry = 19; // The default centry is 19

    week_day = centry / 4 - 2 * centry;
    week_day = week_day + temp_year + temp_year / 4;
    week_day = week_day + 26 * (temp_month + 1) / 10 + day - 1;
    week_day = week_day % 7;

    cout << "The week day is " << week_day << " of ";
    cout << year << "/" << month << "/" << day << "\n";
}

```

2. 给出以下程序的输出结果

```

#include <iostream.h>
class WINDOW {
    public:
        WINDOW() { count = count + 1; }
        ~WINDOW() { count = count - 1; }
        int getwin() { return count; }
    private:
        static int count;
};

```

```

int WINDOW::count = 0;
class WORKWIN: public WINDOW {
public:
    WORKWIN()
    {   cout << "Open a work window" << endl; }
    ~WORKWIN()
    {   cout << "Close a work window" << endl; }
};
class MSGWIN: public WINDOW {
public:
    MSGWIN()
    {   cout << "Open a message window" << endl; }
    ~MSGWIN()
    {   cout << "Close a message window" << endl; }
};
class SCREEN {
public:
    SCREEN(): msgwin(), workwin()
    {   cout << "Initialize the screen" << endl; }
    ~SCREEN()
    {   cout << "Clear the screen" << endl; }
    int getwin()
    {   return workwin.getwin(); }
private:
    WORKWIN    workwin;
    MSGWIN     msgwin;
};
void main()
{   SCREEN    screen;
    WINDOW    msgwin;
    cout << "There are " << screen.getwin() << " window(s) on screen" << endl;
    cout << "There are " << msgwin.getwin() << " message window(s)" << endl;
}

```

四、程序填空（共 24 分，每空 3 分）

根据以下各小题的描述，把相应题号的答案填到答题框中：

1. 在以下程序中，希望对成员函数 void who() 实现动态绑定，请填相应的语句

```

#include <iostream.h>
class BASE{
public:
    ① {cout<<"BASE\n";}
};
class DERIVED:public BASE{
public:
    void who(){cout<<"Derivation\n";}
};
void main()
{   ② *p;        // 定义指针 p
    DERIVED  obj;
    ③
    p->who();      // 此时调用的是 DERIVED 类中定义的 who()
}

```

2. 在以下程序中，函数 fill(int square[MAX_LEN][MAX_LEN], int max_num) 将数字 1, 2, ..., max_num, ..., max_num² 按逆时针填入 max_num×max_num 的矩阵中，下面是 max_num = 5 和 max_num = 6 时主程序的输出结果：

max_num = 5					max_num = 6					
1	16	15	14	13	1	20	19	18	17	16
2	17	24	23	12	2	21	32	31	30	15
3	18	25	22	11	3	22	33	36	29	14
4	19	20	21	10	4	23	34	35	28	13
5	6	7	8	9	5	24	25	26	27	12
					6	7	8	9	10	11

请完成该函数中的空白，使程序运行能得到正确结果。

```
#include <iostream.h>
#include <iomanip.h>
const int MAXLEN = 20;
void fill(int square[MAXLEN][MAXLEN], int maxNum)
{
    int currNum;
    int currLevel;
    int row, col; //row是行号, col是列号
    currNum = 1;    currLevel = 1;
    do { //逆时针生成一圈的矩阵数据
        row = currLevel; col = currLevel;
        do {
            square[row][col] = currNum; row++; currNum++;
        } while (____④____);
        do {
            square[row][col] = currNum; col++; currNum++;
        } while (____⑤____);
        do {
            square[row][col] = currNum; row--; currNum++;
        } while (____⑥____);
        do {
            square[row][col] = currNum; col--; currNum++;
        } while (____⑦____);
        currLevel++;
    } while (currLevel <= (maxNum / 2));
    if ((maxNum % 2) != 0) square[maxNum/2+1][maxNum/2+1] = currNum;
    return;
}
void print(int square[MAXLEN][MAXLEN], int maxNum) //输出矩阵
{
    int row, col;
    for (row = 1; row <= maxNum; row++) {
        for (col = 1; col <= maxNum; col++)
            cout << setw(4) << ____⑧____;
        cout << "\n";
    }
    return;
}
void main()
{
    int square[MAXLEN][MAXLEN];
    int maxNum;
    cout<<"Please input the max num of matrix: ";
    cin>>maxNum;
    fill(square, maxNum);
    print(square, maxNum);
}
```

五、程序设计 (21 分)

1. (13 分) 设计一个股票类 STOCK，实现分析股票走势的功能，并提供简单的主函数使用该类。

描述：股票类 STOCK 中记录了股票上市以来每日收市价的资料（所有股票上市至今都不超过 5000 天），并提供一个分析股票走势的功能：找出该股票连续上涨（包括与前一交易日价格持平）的最多交易天数，并输出这次涨势的每一交易日的价格。请编写简单的主函数使用该类创建一个股票，演示其分析股票走势的功能。

提示：股票的价格以 *float* 类型表示，不关心股票交易日的日期信息；每日收市价资料可依时间次序存放在一个数组 *prices* 中；如果数组中有两段连续上涨的交易天数是一样长，则随便挑选一个。程序获取股票价格的简单方法是从键盘输入，以一个非正常价格表示结束（如 0 或负数）。

要求：

- 1). 从文本文件 in.data 中读入一股票上市以来每日的收市价（所有股票上市至今都不超过 5000 天）；
- 2). 将分析结果，输出到文本文件 out.data；
- 3). 上述指定文件名不得更改；
- 4). 请尽量多提供注释。

2. （8 分）根据下面公式，设计一个计算 π 的程序：

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + \frac{1}{4n-3} - \frac{1}{4n-1}$$

《完》