

浙江工业大学
2018 年硕士研究生招生考试试题

考试科目: 958C++程序设计(II) 共 5 页
★★★★ 答题一律做在答题纸上, 做在试卷上无效。 ★★★★★

一、程序阅读题(共 5 题, 每小题 9 分, 共 45 分)
说明: 请写出下面程序的运行结果。

1.

```
#include<iostream>
using namespace std;
int main() {
    char ch;
    char title[] = "Titanic";
    ch = title[1];
    title[3] = ch;
    cout << title << endl;
    cout << ch << endl;
}
```
2.

```
#include<iostream>
using namespace std;
void f(int x)
{
    x = x+5;
    if (x%2 ==0) x += 6;
    if (x%3 ==0) x += 7;
    if (x%5 ==0) x += 8;
}
int main(void)
{
    int x=5;
    f(x);
    cout <<x <<endl;
}
```
3.

```
#include<iostream>
#include<string.h>
using namespace std;
class Film {
    char* a;
    int b;
public:
    void Init(char* pa, int ib) {
        a=new char[strlen(pa)+1];
```

```

        strcpy(a, pa);
        b=ib;
    }
    char* Geta() {return a;}
    int Getb() {return b;}
    void Output() {cout<<a<<' '<<b<<endl;}
} dx;
int main() {
    Film dy;
    dx.Init("LA LA LAND", 30);
    dy.Init("Robot", 3*dx.Getb()+5);
    dx.Output();
    dy.Output();
}

```

注：忽略字符串常量到 const char* 的转换警告。

4. 假设用户依次输入 1, 2, 2, 5.

```

#include<iostream>
using namespace std;
struct Fract {
    int num;
    int deno;
};
Fract sum(Fract, Fract);
int main()
{
    int num1, deno1, num2, deno2;
    cout<<"请输入分数 1: 分子 分母:";
    cin>>num1>>deno1;
    cout<<"请输入分数 2: 分子 分母:";
    cin>>num2>>deno2;
    Fract f1={num1, deno1}; /* 1/2 */
    Fract f2={num2, deno2}; /* 2/5 */
    Fract result = sum(f1, f2); //sum the fractions
    cout<<result.num<<"/"<<result.deno; //display the result
    return 0;
}
Fract sum(Fract f1, Fract f2)
{
    Fract result= {(f1.num * f2.deno) + (f2.num * f1.deno), f1.deno * f2.deno};
    return result;
}

```

5.

```

#include <iostream>
using namespace std;

```

```

class student
{
    public:
    int rno , m1 , m2 ;
    void get()
    {
        rno = 15, m1 = 10, m2 = 10;
    }
};
class sports
{
    public:
    int sm;
    void getsm()
    {
        sm = 10;
    }
};
class statement:public student,public sports
{
    int tot,avg;
    public:
    void display()
    {
        tot = (m1 + m2 + sm);
        avg = tot / 3;
        cout << tot;
        cout << avg;
    }
};
int main()
{
    statement obj;
    obj.get();
    obj.getsm();
    obj.display();
}

```

二、程序改错题（共 2 题，第 1 小题 5 分，第 2 小题 10，共 15 分）

1. 该程序运行有问题，请改正 main 函数中的错误，使程序的输出结果是：

```

num = 1
num = 10
num = 100

```

注意：不能增行或删行，也不能更改程序的结构。

```

1 #include <iostream>
2 using namespace std;

```

```

3 class TestClass {
4 public:
5 TestClass(int i){
6     number=i; }
7 void SetNumber(int m){
8     number=m; }
9 int GetNumber()const{
10     return number; }
11 void Print() const{ cout<<"number="<<number<<endl; }
12 private:
13     int number; };
14 int main(){
15     TestClass obj1;
16     obj1.Print();
17     TestClass obj2(3);
18     obj1.number = 10;
19     TestClass.SetNumber(100);
20     obj1.Print();
21     obj2.Print(); }

```

```

2. #include <iostream>
class A {
    int i;
public:
    virtual void fun()=0;
    A(int a) {i=a;}
};
class B:public A {
    int j;
public:
    void fun() {cout<<"B::fun() \n"; }
    B(int m, int n=0):A(m), j(n) {}
};
int main() {
    A *pa;
    B b(7);
    pa=&b;
}

```

三、编程题(共 3 题, 每题 30 分, 共 90 分)

1. 编写一个名为 MatrixTranspose 的函数来实现多维数组的转置。
该函数的声明如下:

```

void MatrixTranspose (int a [] [COLUMN_SIZE], int Ta [] [COLUMN_SIZE],
    int rowSize)

```

2. 编写一个名为 "sum_from_to" 的函数, 它带有两个整数参数, 称之为 "first" 和 "last", 并返回 first 和 last 之间(含 first 和 last 本身)的所有整数的和。
例如:


```
cout << sum_from_to (4,7) << endl; //将打印 22, 因为 4+5+6+7=22
cout << sum_from_to (-3,1) << endl; //将打印-5, (-3)+(-2)+(-1)+0+1=-5
cout << sum_from_to (7,4) << endl; //将打印 22, 7+6+5+4=22
cout << sum_from_to (9,9) << endl; //将打印 9
```

3. 本程序将重复提示你输入一个正整数。每次你输入一个正整数, 程序将打印所有它的因子, 因子降序打印, 每个因子占一行。要求因子分解功能用单独的类实现。
请输入一个正整数: 36

36
18
12
9
6
4
3
2
1

你想看尝试另一个整数吗(y/n)? y

请输入一个正整数: -44

-44 不是一个正整数。

请输入一个正整数: 0

0 不是一个正整数。

请输入一个正整数: 109

109

1

你想看另一个整数(Y / N) 的除数吗? m

请以 Y (或 y) 回答是, N (或 n) 回答为否。

你想看另一个整数(Y / N) 的除数吗? n