上 海 交 通 大 学 试 卷（ A 卷）

（ 2017 至 2018 学年 第1学期 ）

班级号\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 学号\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 姓名

课程名称 程序设计思想与方法 成绩

1. **选择题，每题只有一个答案正确（每题1分，共25分）。**

**1. int a = 0x123; cout << a<<endl; 的结果是 。**

**A．291 B. 123 C. 96 D. 语法错误**

**2. 在一个函数中 是错误的。**

**A. 函数中可以没有return语句 B. 函数中如果有返回值，就一定要有return语句.**

**C. 函数中一个return语句可返回多个值给主调函数 D. 函数中可以使用多个return语句**

**3. 若有int a[10][5]，则 可得到该数组最后一个元素**

**A. a[10][5] B. a[9][5] C. \*((\*(a+9))+4) D. (\*(a+4))[9]**

**4. 逻辑运算符&&, ||和！的优先级从高到低依次为**

**A. &&>||>! B !>&&>|| C. !>||>&& D. ||>!>&&**

**5. 构造函数的返回类型是**

**A.void B.取决于类的情况 C.所定义类 D.没有返回类型**

**6. 字符串“1234567890”需要 字节的存储空间**

**A. 10 B. 11 C. 40 D. 44**

**7. 如果要支持Rational到double的类型转换，需要在Rational类中声明并定义 成员函数。**

**A. double operator double() B. double operator double(Rational r)**

**C. operator double() D. operator double(const Rational &r)**

**8. 以下错误的语句是**

**A. int a(5); B.int a[5]; C. int 5a; D. int a5**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 题号 |  |  |  |  |
| 得分 |  |  |  |  |
| 批阅人(流水阅  卷教师签名处) |  |  |  |  |

**我承诺，我将严格遵守考试纪律。**

**承诺人：**

**9. 下列有关类的常对象说法不正确的是**

1. **常对象能调用任何成员函数 B. 常对象是类的一个实例**
2. **对象是类的一个实例 D. 常对象不能改变其属性的值**

**10. 对于任意一个类A，其拷贝构造函数的参数为**

|  |  |  |  |
| --- | --- | --- | --- |
| **A．A a** | **B．A &a** | **C．const A a** | **D．const A &a** |

**11. 关于静态成员函数，不正确的是**

1. **不含有隐含参数this**
2. **只能访问静态类型的属性**
3. **不能通过对象来调用**
4. **能访问const类型的属性**

**12. 在 情况下，一个类必须重载赋值和拷贝构造函数**

1. **类的属性成员中含有静态数组**
2. **类的属性成员中含有指针类型的变量**
3. **类的属性成员中含有const成员**
4. **类的属性成员中含有静态成员**

**13. 下列运算符在C＋＋里不能重载的是**

1. **<< B. || C. /= D. ::**

**14. 语句Rational r1, r2(1,4), r4=4, \*a[2]; 中调用了 次Rational的构造函数?**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A．3** | **B．2** |  | **C．5** | **D．4** |

**15. 在下列各函数中， 不能用类的成员函数来定义**

1. **拷贝构造函数 B. 析构函数 C. []运算的重载 D. <<运算的重载**

**16. Array<int> arr(10,20); 说明arr是**

1. **一个类模板的实例 B. 一次返回模板类的函数调用**
2. **一次返回类模板的函数调用 D. 一个模板类的实例**

**17. 下面关于类的构造函数， 概念是错误的。**

**A． 构造函数是当一个对象被创建时自动调用的 B. 构造函数不可以重载**

**C． 构造函数可以有参数，也可以没有 D. 构造函数的参数可以有缺省值**

**18. 下列函数能够并存的有**

1. **int Fun(int x = 1, int y = 2); int Fun(int x, int y);**
2. **int Fun(int x, int y = 2); int Fun (double x, int y);**
3. **int Fun(int x, int y = 2); double Fun(int x, int y);**
4. **int Fun(int x = 1, int y); int Fun(int x = 3, int y);**

**19. 在语句char \*s=”hello sjtu”, ss[20]; 之后，运行 语句是错误的。**

**A. s=new char[20]; B. s =”hello!”; C. strcpy(s, “sjtu”); D. s=ss;**

**20. 当enum NUM = {ONE, TWO, THREE}; NUM x=TWO; 则cout<<x<<endl;的运行结果为**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **A. 2 B.TWO C.1 D.”TWO”**    **21. 友元运算符obj++被C++编译器解释为**  **A．operator++(obj) B．operator++(obj,0) C．obj.operator++( ) D．obj.operator++(0)**  **22. 下列选项中,全部都是C++关键字的选项为**  **A．while IF Static B．double while integer**  **C．sizeof Static extern D．continue char const**  **23. class A {public: int x; }; A a, \*p; p=&a; 现在要给对象a的属性x赋值为5， 语句正确。**  **A．p->x =5 B. p.x = 5; C. a.->x=5 D. a->x=5;**  **24. 关于操作符重载， 是正确的。**  **A．可以改变操作符的优先级，但不能改变它的结合性**  **B. 可以改变操作符的操作数的个数，但不能改变它的优先级**  **C．可以改变操作数的结合性，但不能改变它的操作数个数**  **C．操作数的优先级、结合性、操作数个数都不能改变**   |  |  | | --- | --- | | **25. 关于类中标为private的属性成员的访问， 是错误的。**  **A． 类的成员函数可以直接访问 B. 类的友元函数可以直接访问**  **C． 任何函数可以直接访问 D. 可以通过该类公有的函数去访问** |  | |  |  |  |  |

1. **给出下列程序的运行结果（每题3分，共18分）。**

**1.**

**#include <iostream>**

**using namespace std;**

**class Point {**

**private: int x, y;**

**public:**

**Point(int a = 1, int b = 1)**

**{ x = a; y = b;**

**cout << "Point(" << x << "," << y << ")" << endl; }**

**~Point() { cout << "~Point(" << x << "," << y << ")" << endl; }**

**};**

**int main( )**

**{ Point \*pp = new Point(3, 4);**

**{ Point point1(-5, 7); }**

**Point point2(1, 2);**

**delete pp;**

**return 0;**

**}**

**2.**

**#include <iostream>**

**#include <cstring>**

**using namespace std;**

**int main()**

**{**

**char \*str[3]={"morning","afternoon","apple"};**

**char \*temp;**

**int i,j;**

**for(i=0;i<3;i++)**

**for(j=i+1;j<3;j++)**

**if(strcmp(str[i],str[j])>0)**

**{**

**temp=str[i];**

**str[i]=str[j];**

**str[j]=temp;**

**}**

**for (i=0;i<3;i++)**

**cout<<str[i]+i<<endl;**

**return 0;**

**}**

**3.**

**#include <iostream>**

**using namespace std;**

**class A**

**{**

**private:**

**int data;**

**public:**

**A(int i=0){data=i;}**

**void setdata(int i) {data=i;}**

**void display() {if (data%2==0) cout<<data<<endl;}**

**};**

**int main()**

**{**

**A a[10];**

**int i,data;**

**for(i=0;i<10;i++)**

**{**

**data=i\*2-i; a[i].setdata(data/2);**

**}**

**for(i=0;i<10;i+=2)**

**a[i].display();**

**return 0;**

**}**

**4.**

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**int score[6]={90,70,88,92,55,68};**

**for (int i=0; i<6; i++)**

**switch(score[i]/10)**

**{**

**case 9: cout << "A";**

**case 8: cout << 'B' <<endl; break;**

**case 7: cout << "C";**

**default: cout << 'D' <<endl;**

**};**

**return 0;**

**}**

**5.**

**#include <iostream>**

**using namespace std;**

**void f(int a, int &b)**

**{**

**static int c=3;**

**c+=c\*2+3;**

**b=a++;**

**}**

**int main()**

**{**

**int a[10]={3,15,9,8,12,35};**

**int b[40]={0};**

**for (int i=0; i<3; i++)**

**{**

**f(a[i], b[i]);**

**b[i]++;**

**cout<<b[i]<<endl;**

**}**

**return 0;**

**}**

**6.**

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**bool y1,y2,y3;**

**int x1=3,x2=4, x3=8;**

**y1="sjtu"||(x1-=6);**

**y2="rose"&&(x2+=2);**

**y3=5&&!(6/5)||(x3=0);**

**cout << x1<<endl;**

**cout << x2<<endl;**

**cout << x3<<endl;**

**return 0;**

**}**

**三、程序填空（每空2分，共22分）。**

**1.将文件clients.txt中的账户内容在屏幕上输出**

**void outputLine( int account, const char\* name, double balance )**

**{ cout<< account << ‘\t’ << name << ’\t’ << balance << endl; }**

**int main()**

**{**

**if ( ) {     cerr << "File could not be opened\n";    exit( 1 );  }**

**int account;**

**char name[ 30 ] ;**

**double balance;**

**cout << "Account" << ‘\t’<< "Name" <<’\t’<< "Balance”<<endl;**

**while ( )**

**outputLine( account, name, balance );**

**return 0;**

**}**

**2. .考试成绩统计**

**int main()**

**{**

**int \*score, num, i，sum;**

**double average;**

**cout << "请输入参加考试的人数：";**

**cin >> num;**

**cout << "请输入成绩：\n";**

**sum=**

**for (i = 0; i < num; ++i)**

**{**

**cin >> score[i];**

**sum+=score[i];**

**}**

**cout << "平均分是：" << average << endl;**

**return 0;**

**}**

**3. void Hanoi(int n, char start, char finish, char temp)**

**{**

**if ( ) cout << start << "->" << finish << '\t';**

**else**

**{**

**cout << start << "->" << finish << '\t';**

**}**

**}**

* 1. **编程题(共35分)。**
     1. **求1+1/2!+1/3!+…+1/n!，要求整个过程加法乘法和除法各自的次数都不能超过n次。（5分）**
     2. **回文数是指无论顺读还是倒读都是一样的数，诸如12321，45677654等。编写一个函数判断由参数传进的一个正整数是否为回文数。如果是，输出“Yes”，否则输出“No”。要求函数用递归方法来实现。**

**#include <iostream>**

**using namespace std;**

**void huiwen(int n);**

**int main()**

**{**

**int n;**

**cout<<"n=";**

**cin>>n;**

**huiwen(n);**

**return 0;**

**}**

**编写函数void huiwen(int n) （15分）**

* + 1. **编写一个程序，定义一个银行系统的账户类SavingAccount，账户含的信息有账号、账户余额。再编写一个main函数，创建两个账号，为两个账号分别存入100，150元，最后利用类中的输出重载输出两个账号的信息（包括账号和余额）。 要求账户中账号自动生成，第一个生成的账号为1，第二个生成的账号为2，依次类推。账户所需的操作包括存钱、<<操作重载以显示账户信息。（15分）**