

## 2004 C++ 试题

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

一	二	三	四	五	总分

一、选择题：从每题给出的四个答案中，在 ABCD 中选择一个正确答案，填入【      】。

1、有以下程序

```
main () {  
    int a=0, b=1, n;  
    cin>>n;  
    do {  
        a++;  
        b--;  
    } while (b!=n);  
    cout<<a<<endl;  
}
```

若要使程序的输出值为 2，则应该从键盘给 n 输入的值是【      】

A.-1    B.-3    C.-5    D.0

2、若有如下程序段，其中 s、a、b、c 均已定义为整型变量，且 a、c 均已赋值（c 大于 0）

```
s=a;  
for (b=1;b<=c;b++)  
    s=s+1;
```

则与上述程序段功能等价的赋值语句是【      】

A.s=a+b;    B.s=a+c;    C.s=s+c;    D.s=b+c;

3、要求以下程序的功能是输出整形数组 a 中最大的元素

```
main () {  
    a[]={1,5,2,8,4,3,6};  
    int m = a[0];  
    for (int i=1; i<sizeof(a); i++)  
        if (m>a[i])  
            m = a[i];  
    cout<<m<<endl;  
}
```

程序运行后输出结果错误，导致错误结果的程序行是【      】

A. int m = a[0];    B. for (int i=1; i<sizeof(a); i++)    C. if (m>a[i])    D. m = a[i];

4、下列选项中正确的语句组是【      】

A. char s [8] ; s= {"Hello"} ;

B. char \*s; s= {"Hello"} ;

C. char s [] ="Hello";

D. char \*s; s="Hello";

5、有以下程序段

```
main () {
```

```
    int a=5, *b, **c;
```

```
    c=&b;
```

```
    b=&a;
```

```
    ...
```

```
} 程序在执行了 c=&b;b=&a;语句后，表达式:**c 的值是 【    】
```

A.变量 a 的地址    B.变量 b 中的值    C.变量 a 中的值    D.变量 b 的地址

6、分析下面的程序

```
class A{
```

```
public:
```

```
    virtual void print(){ cout<<"A"; }
```

```
};
```

```
class B :public A{
```

```
public:
```

```
    void print(){ cout<<"B"; }
```

```
};
```

```
void main(){
```

```
    A a, *pa;    B b, *pb;
```

```
    pa=(A*)&b;    pa->print();
```

```
    pb=(B*)&a;    pb->print();
```

```
}程序的运行结果为 【    】
```

A. 编译错误    B. AB    C. BA    D. AA

7、在 Windows 环境下，分析程序的输出结果 【 】

```
main(){
```

```
    char buf1[20]="Hello";
```

```
    char buf2[20]="world";
```

```
    if (buf1>buf2)
```

```
        cout<<buf1<<endl;
```

```
    else
```

```
        cout<<buf2<<endl;
```

```
}
```

A. 编译错误    B. Hello    C. world    D. 结果不确定

8、分析下面的程序，运行结果为 【 】

```
main(){
```

```
    int a[]={1,2,3,4,5,6,7,8,9};
```

```
    int *p=a;
```

```
cout<<*p++;  
cout<<*(p+1);  
}
```

A. 1 3      B. 2 2      C. 1 2      D. 2 3

9、下面函数中能够体现面向对象的多态性的是【      】

A. 构造函数      B. 虚函数      C. 析构函数      D. 友元函数

10、构造一个类 A，在该类中的将“+”操作符重载为友元函数，则下面语法正确的是【      】

A. friend A operator +=(A&, A&)  
B. friend A& operator +=(A ,A)  
C. friend A& operator +=(A&, A&)  
D. A& operator +=(A&)

11、下面关于数组作为参数传递的函数原型，错误的是【      】

A. void func(int arr[]);  
B. void func(int arr[][4]);  
C. void func(int \*arr);  
D. void func(int arr[4]);

12、下列哪些函数有返回值【      】

A. 转换运算符函数      B. 构造函数      C. 析构函数      D. 赋值运算符函数

13、对于下面的代码，哪一个和 a[i]不等价【      】

```
int a[10];  
int *ptr=a;
```

A. \*(a+i)      B. \*(ptr+i)      C. \*ptr+i      D. ptr[i]

14、多继承的析构顺序是【      】

A. 类自己→内嵌对象→ 虚拟基类→ 非虚拟基类  
B. 类自己→内嵌对象→ 非虚拟基类→虚拟基类  
C. 非虚拟基类→虚拟基类→内嵌对象 →类自己  
D. 虚拟基类→非虚拟基类→内嵌对象 → 类自己

15、如果让类的所有成员在类的范围内共享某一个函数，则声明该函数为【      】

A. static      B. friend      C. virtual      D. const

16、以下对于析构造函数说法中，正确的是【      】

A. 每个类必须自己定义一个析构函数      B. 析构造函数需要在程序中调用  
C. 一个类中只能定义一个析构造函数      D. 析构造函数可以有形式参数

17、下面哪一个变量占用了内存空间【      】

A. enum COLOR{RED ,BLUE, GREEN}; 中的 COLOR  
B. int a; int &r=a; 中的变量 r

- C. `const int *p;`中的 `p`  
D. `struct Student`  
{  
    `int NO;`  
    `char name[20];`  
};中的 `Student;`

18、构造一个类时，类中有些函数系统缺省会进行构造，那么下面哪一类系统不会产生默认  
的函数 【      】

- A. 构造函数    B. 友元函数    C. 拷贝构造函数    D. 赋值运算符

19、在类的继承过程中，哪些函数能够被继承 【      】

- A. 虚函数      B. 构造函数    C. 析构函数    D. 友元函数

20、下面程序的输出结果是 【      】

```
void main(){  
    int a=1,b=10;  
    int &r = a;  
    int *p = &b;  
    *p=r++;  
    cout<<a<<" "<<b<<endl;  
}
```

- A. 1,10    B. 2,10    C. 1,2    D. 2,1

## 二、填空题

- 通常 C++ 的算法和程序结构有\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_三种。
- 在 C++ 中有二种参数传递方式：值传递和\_\_\_\_\_。
- 重载函数的特征\_\_\_\_\_。
- 当建立一个新对象时，程序自动调用\_\_\_\_\_；当销毁一个对象时程序自动调用\_\_\_\_\_。
- 如果类名 A，则该类的拷贝构造函数原型为\_\_\_\_\_。
- 定义类的成员时，有哪三种访问控制修饰符\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。
- 至少包含一个\_\_\_\_\_的类称为抽象类。
- 通常情况下，内存被分成\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_4 部分，其中，`const` 变量保存在\_\_\_\_\_，局部变量保存在\_\_\_\_\_。
- 对象的三大基本特性是\_\_\_\_\_、\_\_\_\_\_、封装性。

## 三、分析下列程序，写出程序的运行结果

```
1、void main()  
{  
    int n=1;  
    for (int i=1;i<11;i+=n)  
    {
```

```

        n++;
        if (i % 3==2)
            continue;
        cout<<i<<endl;
    }
}

```

---



---



---



---

2、void fn(int b)

```

{
    static int a=10;
    cout<<a--<<" "<<++b<<endl;
}
void main()
{
    for (int i=0;i<4;i++)
        fn(i);
}

```

---



---



---



---

#### 四、补充下面的程序

1、已知整型数组 a 中的元素按照从小到大排列，现任意向数组中插入一整数，使得该数组依然保持从小到大排列。该函数原型如下，请将其补充完全。

```

void insert(int x, int a[],int len)
{
    for (int i=0;i<len;i++)
        ____ [1] ____ //获取 x 插入的位置
        break;
    for (int j=len;j>i;j--)
        ____ [2] ____ ; //将 x 插入位置的数组元素后移
        ____ [3] ____ ; //将 x 插入该位置
}

```

2、下面的函数将字符串 s 中字符逐一颠倒，该函数使用指针 t 指向字符串最后一个字符，然后首尾字符相互交换的方法实现，比如 s = “Hello”，经过该函数后 s= “olleH”，将下面的算法补充完全。

```

void reverse(char *s)
{
    char c;

```

```

char *t;
for (t = s + strlen(s) - 1; ___[1]___;t--,s++)
{
    ch = *s ;
    ___[2]___;                //字符首尾字符交换
    ___[3]___;
}
}

```

3、以下的程序将输入的十进制整数  $n$  转换为  $b$  进制数( $b \leq 8$ )，并将转换结果输出。转换的方法是"除  $b$  取余"。

```

void transfer(int m,int k)
{
    int a[20];
    for (int i=0;m>0;i++)
    {
        a[i]=___[3]___;
        m/=k;
    }
    ___[4]___;
    cout<<a[j];
}
void main()
{
    int b,n;
    cout<<"请输入十进制数 n 和进制数 b\n";
    ___[1]___;
    ___[2]___;
}

```

五、分析下面的程序，按要求将完成下列程序。

```

1、#include <string.h>
class A{
public:
    A(char *pName="noname")
    {
        name=new char[strlen(pName)+1];
        strcpy(name,pName);
    }
protected:
    char *name;
};
void main()
{
    A a;
    A b(a);
}

```

```
}
```

1) 程序运行中会产生内存泄漏, 为什么?

2) 请添加一个析构函数, 避免 1) 中的问题

3) `main()` 函数中调用了拷贝构造函数, 当只添加析构函数时, 程序会出现什么问题?  
添加一个拷贝构造函数, 避免该问题。

2、`#include <iostream.h>`

```
class A{
```

```
public:
```

```
    A(){}
```

```
    A operator ++(int)
```

```
    {
```

```
        A temp(*this);
```

```
        a++;
```

```
        return temp;
```

```
    }
```

```
    void display() { cout<<a<<endl;}
```

```
protected:
```

```
    int a;
```

```
};
```

```
void main(){
```

```
    A a;
```

```
    (++a).display();
```

```
    a++.display();
```

```
}
```

1) 该程序编译出错, 请判断程序出错的位置

2) 将该程序补充完全, 使得编译正确。

3) 如果要使程序的运行结果为 1 和 1, 那么程序还需要如何改动?

答案

一、

DBCCCCBABCDDCBACCBAD

二、

地址传递

函数名称相同，参数类型或个数不同

构造函数、析构函数

A(A& a)

public、protected 、 private

纯虚函数

堆区 栈区 全局数据区 代码区 全局数据区 栈区

多态 继承

三、

1 3 6 10

10,1 9,2 8,3 7,4

四、

```
1  if (x<a[i])
    a[j]=a[j-1];
    a[i]=x;
2  t<s
    *s=*t;
    *t=ch;
3. cin>>n>>b;
    transfer(n,b);
    m % k
    for(int j=i-1;j>=0;j--)
```

五

1、

1) 对象在构造时申请了堆内存，但析构函数并没有释放，因此造成内存泄漏。

```
2) ~A(){
    delete name;
}
```

3) b 和 a 共用了同一段内存，调用析构函数时，同一段内存释放了两次，出现内存错误。这是浅拷贝带来的问题。

```
A(A& a){
    name=new char[strlen(a.Name)+1];
    strcpy(name, a.Name);
}
```

2、

1) (++a).display();出错

```
2) A& operator ++ (){
    a++;
```



```
        return *this;  
    }  
3) 构造函数改为:  A():a(0){}
```

填空题:

1. enum struct union
2. 顺序, 循环, 选择
3. .cpp .h 头 源
4. 变量起作用的范围 块作用域 文件作用域 函数作用域 类作用域
5. sizeof() sizeof(对象名)
6. 当前活动
7. 初始化列表//
8. 抽象类
9. 公有继承 私有继承 保护继承
10. 成员 友元
11. 模板 运行 重载
12. 16

单选题

1. D
2. C
3. C
4. D
5. B
6. A
7. D
8. B
9. B
10. B
11. A
12. D
13. C
14. B
15. A
16. C
17. D
18. A
19. B
20. A

简答题:

1. Base class Print :1  
Derived Class Print :12  
Base class Print :3  
Derived Class Print :5  
Derived Class Print :10
2. a 1  
e 2
3. <ifstream> f.open("input.txt") f>>c c!==' ' cout<<c
4. t-s>0 \*s=\*t \*t=ch

5.函数实现了一个求一个数 n 次方的功能 返回值为 81

6. a,b int &i ,int &j t=i i=j j=t

四.

```
X(int x=0) m_x=x; virtual public X Y(int y=0, int t=0):X(t) m_y=y;
```

```
virtual public X Z(int z=0, int s=0):X(s) m_z=z; public Y , public Z
```

```
A(int a=0, int b=0, int c=0, int d=0): Y(b,d) ,Z(c,d), X(d) m_a=a;
```

运行结果： 4

五.

1. 因为程序动态申请了内存空间，但结束时没有释放掉，那块空间没有归还系统，造成内存泄露

2. 在 public 中添加：

```
~A ()
```

```
{ delete [name; }
```

3. 当只添加析构函数时，结束程序时由于两个对象中的指针都指向同一内存地址，而造成无法两次释放该空间的错误，系统运行不正常，会报错  
在 public 中添加拷贝构造函数，

```
A (A&a)
```

```
{
```

```
name=new char[strlen (a.name )+1 ];
```

```
strcpy(name a.name);
```

```
}
```

# 浙江大学“C 程序设计及实验”试题卷

2003-2004 学年春季学期

考试时间: 2004 年 6 月 25 日上午 8:30-10:30

注意: 答题内容必须写在答题卷上, 写在本试题卷上无效

## 一. 单项选择题 (每题 1 分, 共 10 分)

1. C 语言程序的三种基本结构是顺序结构、选择结构和\_\_\_\_\_结构。  
A. 递归                      B. 转移                      C. 循环                      D. 嵌套
2. 下列运算符中, 优先级最高的是\_\_\_\_\_。  
A. +=                      B. []                      C. ?:                      D. &
3. 运算符\_\_\_\_\_不能做为单目运算符。  
A. /                      B. &                      C. \*                      D. +
4. 以下选项中不正确的字符常量是\_\_\_\_\_。  
A. '1'                      B. '\0'                      C. '%d'                      D. 10
5. 表达式  $x \&\&1$  等价于\_\_\_\_\_。  
A.  $x==0$                       B.  $x==1$                       C.  $x!=0$                       D.  $x!=1$
6. 已知字符 'a' 的 ASCII 码为 97, 执行下列语句的输出是\_\_\_\_\_。  
`printf("%c%d", 'b', 'b'+1);`  
A. b99                      B. 98c                      C. 9899                      D. bc
7. 在 C 语言程序中, 在函数内部定义的变量称为\_\_\_\_\_。  
A. 全局变量                      B. 外部变量                      C. 局部变量                      D. 内部变量
8. 执行语句 `printf("_____", "hello");` 将得到出错信息。  
A. %u                      B. %s                      C. %x                      D. %f
9. 对于以下的变量定义, 表达式\_\_\_\_\_是不正确的。  

```
struct node {  
    float x, y;  
} point, *p=&point;
```

  
A.  $p \rightarrow x=2.0$                       B.  $(*p).y=3.0$                       C.  $point.x=2.0$                       D.  $*p \rightarrow y=3.0$
10. 对于变量定义: `int **p[10]`, p 是一个\_\_\_\_\_。  
A. 指针                      B. 数组                      C. 函数                      D. 数组元素

## 二、填空题 (每题 2 分, 共 30 分)

1. 写出计算  $s(s-a)(s-b)(s-c)$  的 C 语言表达式\_\_\_\_\_。
2. 如果运算符 \* 和 / 都是右结合的, 则表达式  $7 * 2 / 3$  的值是\_\_\_\_\_。
3. 表达式  $!!10$  的值是\_\_\_\_\_。
4. 表达式  $\sim(\sim 1 < 1)$  的值是\_\_\_\_\_。
5. 下列程序段的输出结果是\_\_\_\_\_。  

```
#define MM(x,y) (x)+(y)  
printf("%d", MM(2*3,15)*2);
```

6. 对于变量定义：int a[10], \*p=a+3; 数组 a 的最后一个元素是 p[\_\_\_]。

7. 下列程序段的输出结果是\_\_\_\_\_。

```
int k=1, j=2;
int *p=&k, *q=&j;
(*p)++;
k=j;
(*p)++;
printf("%d, %d", k, j);
```

8. 下列程序段的输出结果是\_\_\_\_\_。

```
char *st[ ]={"ONE", "TWO", "FOUR", "K"};
printf("%s, %c\n", *(st+1)+1, (*(st+2)+2)-1);
```

9. 下列程序的输出结果是\_\_\_\_\_。

```
#include <stdio.h>
void p(int *x, int *y)
{ int *p;
  p=x; x=y; y=p;
}
void main()
{ int x=0, y=3;
  p(&y, &x);
  printf("%d, %d", x, y);
}
```

10. 用 typedef 定义一个整型指针类型 PA。

### 三、程序阅读题（每题5分，共30分）

1. 输入 2 59 60 <回车>后，下列程序的输出结果是\_\_\_\_\_。

```
#include <stdio.h>
void main( )
{ int ri, repeat;
  int mark;
  scanf("%d", &repeat);
  for(ri=1; ri<=repeat; ri++)
    scanf("%d", &mark);
    if(mark >= 60)
      printf("Pass");
    else
      printf("Fail");
}
```

2. 输入 2+5\*3/2\$= <回车>后，下列程序的输出结果是\_\_\_\_\_。

```
#include <stdio.h>
void main( )
```

```

{   int op1, op2, res;
    char operator;
    scanf("%d", &op1);
    operator = getchar();
    while(operator!='='){
        scanf("%d", &op2);
        switch(operator){
            case '+': res = op1+op2; break;
            case '-': res = op1-op2; break;
            case '*': res = op1*op2; break;
            case '/': res = op1/op2; break;
            default: res = 0;
        }
        op1 = res;
        printf("%d#", res);
        operator = getchar();
    }
}

```

3. 调用函数 f(2004, 4, 10)的返回值是\_\_\_\_\_。

```

int f(int x, int y, int z)
{   int k, flag;
    int tab[2][13]={
        {0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},
        {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}
    };

    flag=x%4==0 && x%100!=0 || x%400==0;
    for(k=1; k<y; k++)
        z += tab[flag][k];
    return z;
}

```

4. 调用函数 f(15)的输出结果是\_\_\_\_\_。

```

void f(int n)
{   if(n<3)
    printf("%d", n);
    else{
        f(n/3);
        printf("%d", n%3);
    }
    return;
}

```

```
}
```

5. 输入 How are you?<回车>后, 下列程序的输出结果是\_\_\_\_\_。

(函数 strcmp(s, t)的功能: 若 s 和 t 相等,返回 0; 若 s 大于 t,返回一个正数; 若 s 小于 t,返回一个负数)

```
#include <stdio.h>
#include <string.h>
void main( )
{   int i;
    char s[80], ss[80];

    scanf("%s", s);
    strcpy(ss,s);
    for(i=1; i<3; i++){
        scanf("%s", s);
        if(strcmp(s, ss) > 0)
            strcpy(ss,s);
    }
    printf("%s\n", ss);
}
```

6. 下列程序的功能是\_\_\_\_\_。

```
#include <stdio.h>
# include <stdlib.h>
void main()
{   FILE *fpa, *fpb;

    if((fpa=fopen("t.c","r")) == NULL){
        printf("can not open file a.txt !\n");
        exit(0);
    }
    if((fpb=fopen("b.txt","w")) == NULL){
        printf("can not open file b.txt!\n");
        exit(0);
    }
    while(!feof(fpa))
        fputc(fgetc(fpa), fpb);
    fclose(fpa);
    fclose(fpb);
}
```

#### 四、程序(段)填空题 (每空2分, 共40分)

1. 读入 1 个正实数 eps, 计算并输出  $1 - 1/3 + 1/5 - 1/7 + \dots$ , 直到最后一项的绝对值小于 eps。

```

#include <stdio.h>
#include <math.h>
void main( )
{
    int temp, flag;
    double eps, item, sum;

    scanf("%le", &eps);
    sum = 0; item = flag = temp = 1;
    while(fabs(item)>=eps){
        sum = sum + item;
        temp = (1);
        (2);
        item = (3) *flag/temp;
    }
    printf("%f\n", sum);
}

```

2. 输入一行字符，分别统计出其中的英文字母、空格、数字和其他字符的个数。

```

#include <stdio.h>
void main( )
{
    int blank=0, digit=0, letter=0, other=0;
    char c;

    c = getchar();
    while((4)){
        if((5)) /* 判断英文字母 */
            letter++;
        else if((6)) /* 判断数字字符 */
            digit++;
        else if((7)) /* 判断空格 */
            blank++;
        else
            other++;
        (8);
    }
    printf("letter=%d, blank=%d, digit=%d, other=%d\n", letter, blank, digit, other);
}

```

3. a 是一个  $3 \times 3$  的矩阵，输入 a 的元素，如果 a 是下三角矩阵，输出"YES"，否则，输出"NO"。(下三角矩阵，即主对角线以上的元素都为 0，主对角线为从矩阵的左上角至右下角的连线)

```

#include "stdio.h"

```



```

void main( )
{   int a[3][3], flag, i, k;

    for (i=0;i<3;i++)
        for (k=0;k<3;k++)
            scanf("%d",&a[i][k]);
    (9) ;
    for (i=0;i<3;i++)
        for ( (10) ; k<3;k++)
            if( (11) ){
                (12) ;
                break;
            }
    if(flag) printf("YES\n");
    else printf("NO\n");
}

```

4. 定义函数 f(number)，它的功能是返回 number 的逆序数。例如 f(-123)的返回值是-321。

```

long f(long number)
{   int flag, digit;
    long res = 0;

    flag = number<0 ? -1 : 1;
    if(number<0)   number = - number;
    while(number!=0){
        (13) ;
        number /= 10;
    }
    (14) ;
}

```

5. 编写一个函数 delchar(s, c)，该函数将字符串 s 中出现的所有 c 字符删除。

```

void delchar(char s[],char c)
{   int j=0, k=0;

    while(s[j] != '\0'){
        if(s[j] != c){
            s[k] = s[j];
            (15) ;
        }
        (16) ;
    }
}

```

\_\_\_\_\_(17)\_\_\_\_;  
}

6. 请仔细阅读函数 f1, 然后在函数 f2 中填入正确的内容, 使函数 f1 和函数 f2 有相同的功能。

```
void f1(char s[ ], char t[ ])
{   int j, k;
    j=k=0;
    while(s[j]!='\0')
        j++;
    while((s[j++]=t[k++]!='\0'))
        ;
}
```

```
void f2(char *s, char *t)
{
    while(_____(18)____)
        _____(19)____++;
    while(_____(20)____)
        ;
}
```

# 浙江大学“C 程序设计及实验” 参考答案

2003-2004 学年春季学期(2004 年 6 月 25 日)

## 一、单项选择题（每题 1 分，共 10 分）

- |            |            |            |            |             |
|------------|------------|------------|------------|-------------|
| 1 <u>C</u> | 2 <u>B</u> | 3 <u>A</u> | 4 <u>C</u> | 5 <u>C</u>  |
| 6 <u>A</u> | 7 <u>C</u> | 8 <u>D</u> | 9 <u>D</u> | 10 <u>B</u> |

## 二、填空题（每题 2 分，共 20 分）

- |                              |  |
|------------------------------|--|
| 1 <u>s*(s-a)*(s-b)*(s-c)</u> | 2 <u>0</u>                               |
| 3 <u>1</u>                   | 4 <u>3</u>                               |
| 5 <u>36</u>                  | 6 <u>6</u>                               |
| 7 <u>3, 2</u> (错 1 个数字扣 1 分) | 8 <u>WO,T</u> (错 1 项扣 1 分)               |
| 9 <u>0,3</u> (错 1 个数字扣 1 分)  | 10 <u>typedef int * PA;</u> (错 1 项扣 1 分) |

## 三、程序阅读题（每题 5 分，共 30 分）

- |                                 |                                 |
|---------------------------------|---------------------------------|
| 1 <u>Pass</u> (Fail Pass 扣 3 分) | 2 <u>7#21#10#0</u> (错 1 项扣 2 分) |
| 3 <u>101</u>                    | 4 <u>120</u> (错 1 个数字扣 2 分)     |
| 5 <u>you?</u> 少 ? 扣 1 分         | 6 <u>文件复制 a.txt=&gt;b.txt</u>   |

## 四、程序填空题（每空 2 分，共 40 分）

- |   |   |
|---|---|
| (1) <u>temp + 2</u>   | (2) <u>flag = -flag</u>                     |
| (3) <u>1.0 或 (float) 或 (double)</u>   | (4) <u>c!=='\n'</u>                         |
| (5) <u>c&gt;='a' &amp;&amp; c &lt;= 'z'    c&gt;='A' &amp;&amp; c &lt;= 'Z'</u> | (6) <u>c&gt;='0' &amp;&amp; c &lt;= '9'</u> |
| (7) <u>c==' '</u>   | (8) <u>c=getchar()</u>                      |
| (9) <u>flag=1</u>   | (10) <u>k=i+1</u>                           |
| (11) <u>a[i][k]!=0</u>  | (12) <u>flag=0</u>                          |
| (13) <u>res = res*10 + number%10</u>  | (14) <u>return flag*res</u>                 |
| (15) <u>k++</u>   | (16) <u>j++</u>                             |
| (17) <u>s[k] = '\0'</u>   | (18) <u>*s != '\0'</u>                      |
| (19) <u>S</u>   | (20) <u>*s++ = *t++</u>                     |

# 浙江大学 2004-2005 学年冬季学期

## 《C Programming》课程期末考试试卷

开课学院：计算机学院， 考试形式：闭卷

考试时间：2005 年 1 月 22 日，所需时间：120 分钟

考生姓名：\_\_\_\_\_ 学号：\_\_\_\_\_ 专业：\_\_\_\_\_

(注意：答题内容必须写在答题卷上，写在本试题卷上无效)

### Section 1: Single Choice (2 mark for each item, total 20 marks )

1. The precedence of operator \_\_\_\_\_ is the lowest one.  
A . ? :                      B . ==                      C . +=                      D . &
2. \_\_\_\_\_ is correct if it is used as a character constant.  
A . '\ '                      B . '\080'                      C . '%d'                      D . 0xa
3. According to the declaration: char c1=92,c2=92; the value of expression \_\_\_\_\_ is 0.  
A . c1^c2                      B . c1&c2                      C . ~c2                      D . c1|c2
4. According to the declaration: int x=11; the value of expression (x++\*1/3) is \_\_\_\_\_.  
A . 3                      B . 4                      C . 0                      D . 3.667
5. The value of expression sizeof("\num=%d\t") is \_\_\_\_\_.  
A . 7                      B . 8                      C . 9                      D . 10
6. In the following assignments or initialization, \_\_\_\_\_ is wrong.  
A . char s[ ]="hello";                      B . char s[10]; s="hello";  
C . char \*p="hello";                      D . char \*p; p="hello";
7. The following code fragment prints out \_\_\_\_\_.  
#define MA(x, y) (x)\*(y)  
int i = 2;  
i = 3/MA(i, i+2)+5;  
printf("%d\n", i);  
A . 5                      B . 8                      C . 9                      D . 11
8. static struct {  
    int x, y[3];  
} a[3] = {{1,2,3,4},{5,6,7,8},{9,10,11,12}}, \*p;  
p = a+1;  
The value of expression \*((int \*) (p+1)+2) is \_\_\_\_\_.  
A . 3                      B . 7                      C . 10                      D . 11
9. After running the following code fragment, the value of s is \_\_\_\_\_.  
int i=5, s=0;

- do if (i%2) continue; else s+=i; while (--i);  
 A . 15                      B . 9                      C . 6                      D . 5
10. According to the declaration: int (\*p)[10], p is a(n) \_\_\_\_\_.  
 A . pointer                      B . array                      C . function                      D . element of array

**Section 2: Fill in the blanks (2 mark for each item, total 30 marks )**

- The value of expression  $1+4/5+15<7+4\%5+(8,10)$  is \_\_\_\_\_.
- The value of expression  $!!10$  is \_\_\_\_\_.
- The value of expression  $3>2>1$  is \_\_\_\_\_.
- The value of expression  $\sim(-1<<1)$  is \_\_\_\_\_.
- The statement `for (i=1; i<=9; i++) printf("%3d", _____);` prints out the following numbers:  
     1    4    7 10 13 16 19 22 25.
- According to the declaration: `int a[10], *p=&a[1]+2;` the last element of array a is p[\_\_\_\_\_].
- Write the declaration \_\_\_\_\_ with typedef, which makes PA a synonym for a character pointer array, which contains 100 elements.
- The following code fragment prints out \_\_\_\_\_.  

```
static int a[3][4]={{1,2,3},{4,5,6}};
printf("%d",a[0][5]);
```
- The following code fragment prints out \_\_\_\_\_.  

```
char a[]={"678","45"},**p=a+1;
printf("%s,%c",*p,**p-1);
```
- The following code fragment prints out \_\_\_\_\_.  

```
int *p, *q, k = 1, j=10;
p=&j; q = &k ; p = q ; (*p)++;
printf("%d",k);
```
- The following program prints out \_\_\_\_\_.  

```
#include <stdio.h>
void f(int *x,int *y)
{ int *p;
  p=x; x=y; y=p;
}
void main()
{ int x=1, y=2;
  f(&y, &x);
  printf("%d, %d", x, y);
}
```
- The following program prints out \_\_\_\_\_.  

```
#include <stdio.h>
#include <string.h>
main()
{ char st[20]="hello\0world!";
```

- ```

        printf("%d,%d\n",strlen(st),sizeof(st));
    }

```
13. To execute the command: prog 123 456 ABC, the value of `*(&argv[2])` is\_\_\_\_\_.
14. The following program fragment prints out \_\_\_\_\_.
- ```

int i;
int f(int x)
{
    static int k = 0;
    x+=k++;
    return x;
}
i=f(2);
i=f(3);
printf("%d",i);

```
15. The following program fragment prints out \_\_\_\_\_.
- ```

int f(int x)
{
    return ((x>0)? x*f(x-1):3); }
printf("%d",f(f(1)));

```

**Section 3: Read each of the following programs and answer questions ( 5marks for each item, total marks: 30)**

- 1 . The output of the following program is \_\_\_\_\_.
- ```

#include <stdio.h>
void main()
{
    int i,j,k=19;
    while (i=k-1) {
        k-=3;
        if(k%5==0) { i++; continue; }
        else if(k<5) break;
        i++;
    }
    printf("i=%d,k=%d\n",i,k);
}

```
- 2 . When input: **AabD <ENTER>**, The output of the following program is \_\_\_\_\_.
- ```

#include <stdio.h>
void main()
{
    char s[81];
    int i=0;
    gets(s);
}

```

```

while (s[i]!='\0'){
    if(s[i]<= 'z'&& s[i]>= 'a')
        s[i]= 'z'+ 'a'-s[i];
    i++;
}
puts(s);
}

```

3 . The output of the following program is \_\_\_\_\_.

```

#include <stdio.h>
int x,y,z,w;
void p(int x, int *y)
{
    int z;
    ++x;
    ++*y;
    z=x+*y;
    w+=x;
    printf("%2d%2d%2d%2d#", x,*y,z,w);
}
void main()
{
    x=y=z=w=2;
    p(y, &x);
    printf("%2d%2d%2d%2d\n", x,y,z,w);
}

```

4 . The output of the following program is \_\_\_\_\_.

```

#include <stdio.h>
#define F(k)      k+3.14
#define P(a)      printf("a=%d\n", (int)(a))
#define P1(a)      P(a);putchar('\n');
#define P2(a, b)   P(a);P1(b);
void main()
{
    int x = 1;
    {
        int x = 2;
        P(x*F(2));
    }
    {
        for (; x < 10; x += 50)
            P2(x, 9.15*x+32);
    }
}

```

5 . When input: **this is a test.<ENTER>**, The output of the following program is \_\_\_\_\_.

```
#include <stdio.h>
#define TRUE 1
#define FALSE 0
int change(char *c,int status);
void main()
{
    int flag=TRUE;
    char ch;
    do{
        ch=getchar();
        flag=change(&ch,flag);
        putchar(ch);
    } while(ch!='.');
    printf("\n");
}
int change(char *c,int status)
{
    if(*c==' ') return TRUE;
    if(status&&*c<='z'&&*c>='a') *c+='A'-'a';
    return FALSE;
}
```

6 . There are three text files *f1*, *f2* & *f3*, each of them contains some characters as following:

| file name | contents    |
|-----------|-------------|
| <i>f1</i> | <i>aaa!</i> |
| <i>f2</i> | <i>bbb!</i> |
| <i>f3</i> | <i>ccc!</i> |

Compiling the following C source codes, and linking the related object codes, an executable command file *ex12.exe* will be produced. To execute the command at DOS prompt: **ex12 f1 f2 f3<ENTER>**, the output is: \_\_\_\_\_.

```
#include <stdio.h>
main(int argc, char *argv[])
{
    FILE *fp;
    void sub(FILE *);
    int i=1;
    while (--argc>0)
        if ((fp=fopen(argv[i++], "r"))==NULL) {
            printf("Cannot open file!\n");
        }
```



```

        exit(1);
    } else {
        sub(fp);
        fclose(fp);
    }
}

void sub(FILE *fp)
{
    char c;
    while((c=getc(fp))!='\n') putchar(c+1);
}

```

**Section 4: According to the specification, complete each program (2 mark for each blank, total: 20 marks)**

1 . The following program is to calculate the value of “e” according to the formula

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \Lambda, \text{ while the value of the last item must be less than } 10^{-6}.$$

```

#include <stdio.h>
main()
{
    int i;
    double e,item;
    (1);
    item=1.0;
    for (i=1; (2); i++) {
        item/=(double)i;
        e+= (3);
    }
    printf("e=%f\n",e);
}

```

2 . The following program deletes the non-nested comments which be included between /\* and \*/ from the C source program file **exam.c**, and stores the results in the file **exam.out**.

```

#include <stdio.h>
void delcomm(FILE *fp1, FILE *fp2)
{
    int c,i=0;
    while((4)!=EOF)
        if (c=='\n')
            fprintf(fp2, "\n");
        else

```

```

switch(i){
    case 0:
        if(c=='/') i=1;
        else fprintf(fp2,"%c",c);
        break;
    case 1:
        if(c=='*') i=2;
        else {
            fprintf(fp2,"%c",c);
            i=0;
        }
        break;
    case 2:
        if(c=='*') i=3;
        break;
    case 3:
        i=(c=='/')? (5);
        break;
}
}
void main()
{
    FILE *fp1,*fp2;
    fp1=fopen("exam.c","r");
    fp2=fopen("exam.out","w");
    delcomm((6));
    (7);
    return;
}

```

- 3 . Given: the pointer **head** points to the first node of the simple list. The following function **del()** deletes the first node which value is equal to **num** from the simple list.

```

#include <stdio.h>
struct student {
    int info;
    struct student *link;
};
struct student *del(struct student *head,int num)
{
    struct student *p1,*p2;
    if(head==NULL)

```

```
printf("\nlist null!\n");
else {
    p1=head;
    while(__ (8) __) {
        p2=p1;
        p1=p1->link;
    }
    if(num==p1->info){
        if(p1==head)__(9)__;
        else __(10)__;
        printf("delete:%d\n",num);
    } else
        printf("%d not been found!\n",num);
}
return(head);
}
```

# 浙江大学 2004-2005 学年冬季学期

## 《C Programming》课程期末考试答案

开课学院：计算机学院， 考试形式：闭卷

考试时间：2005 年 1 月 22 日，所需时间：120 分

### Section 1: Single Choice (2 mark for each item, total 20 marks)

- |            |            |            |            |             |
|------------|------------|------------|------------|-------------|
| 1 <u>C</u> | 2 <u>D</u> | 3 <u>A</u> | 4 <u>A</u> | 5 <u>B</u>  |
| 6 <u>B</u> | 7 <u>C</u> | 8 <u>D</u> | 9 <u>C</u> | 10 <u>A</u> |

### Section 2: Fill in the blanks (2 mark for each item, total 30 marks)

- |                                |                |
|--------------------------------|----------------|
| 1 <u>1</u>                     | 2 <u>1</u>     |
| 3 <u>0</u>                     | 4 <u>1</u>     |
| 5 <u>3*i-2</u>                 | 6 <u>6</u>     |
| 7 <u>typedef char *PA[100]</u> | 8 <u>5</u>     |
| 9 <u>45,3</u>                  | 10 <u>2</u>    |
| 11 <u>1,2</u>                  | 12 <u>5,20</u> |
| 13 <u>5</u>                    | 14 <u>4</u>    |
| 15 <u>18</u>                   |                |

**Section 3: Read each of the following programs and answer questions ( 5 marks for each item, total marks: 30)**

- 1 i=6,k=4
- 2 AzyD
- 3 3 3 6 5# 3 2 2 5
- 4 a=7#a=1#a=498##
- 5 This Is A Test
- 6 bbbcccd

**Section 4: According to the specification, complete each program (2 mark for each blank, total: 20 marks)**

- |                                    |                                                         |
|------------------------------------|---------------------------------------------------------|
| (1) <u>e=1.0</u>                   | (2) <u>item&gt;=1e-6</u>                                |
| (3) <u>item</u>                    | (4) <u>c=fgetc(fp1)</u>                                 |
| (5) <u>0:2</u>                     | (6) <u>fp1,fp2</u>                                      |
| (7) <u>fclose(fp1),fclose(fp2)</u> | (8) <u>num!=p1-&gt;info&amp;&amp; p1-&gt;link!=NULL</u> |
| (9) <u>head=p1-&gt;link</u>        | (10) <u>p2-&gt;link=p1-&gt;link</u>                     |

# C++程序设计期终考试试题

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

| 一 | 二 | 三 | 四 | 五 | 六 | 总分 |
|---|---|---|---|---|---|----|
|   |   |   |   |   |   |    |

## 一、填空题（每空 1 分）

- 1、 面向对象程序设计的三个主要特性是封装性\_\_\_\_\_和\_\_\_\_\_。
- 2、 类的数据成员是描述对象的\_\_\_\_\_，类的成员函数描述的是对象的\_\_\_\_\_。
- 3、 分数 $\frac{7}{2}$ (值为 3.5)用 C++表达式应该表示为\_\_\_\_\_。
- 4、 如果需要在输出语句中使用格式控制符，必须包含的头文件是\_\_\_\_\_。
- 5、 通过语句 `StuA.setName("Wang");` 可以将 `Student` 类对象 `StuA` 的名字设置为 `Wang`，则类的成员函数 `setName` 的原型是\_\_\_\_\_。
- 6、 在 C++中，执行 `cout<<012+021<<endl;` 所显示的结果是\_\_\_\_\_。
- 7、 在 C++中，表达式 `return 10&&-10;` 的返回值是\_\_\_\_\_。
- 8、 在类的定义中，类的成员默认的控制访问方式是\_\_\_\_\_。
- 9、 在 C++中，如果函数 `Fun` 的形式参数是两个 `int` 类型的引用，返回值也是 `int` 型，这个函数的原型应该写为：\_\_\_\_\_。
- 10、 每个类都有自己的构造函数、拷贝构造函数和析构函数，这三类函数中，不可以指定为虚函数的是\_\_\_\_\_和\_\_\_\_\_。
- 11、 如果 `A1` 是 `Base` 类的一个对象，则执行语句 `Base A2=A1;` 将会自动调用 `Base` 类的\_\_\_\_\_。
- 12、 至少包含一个纯虚函数的类称为\_\_\_\_\_。
- 13、 定义变量是静态类型所使用的关键字是\_\_\_\_\_，说明变量是外部变量的关键字是\_\_\_\_\_。
- 14、 若有 `for` 循环语句 `for(int x=0; x=3; x++){;}`，此循环语句执行的次数将是\_\_\_\_\_次。
- 15、 派生类对象要能够访问基类成员的条件有两个，即\_\_\_\_\_和\_\_\_\_\_。

二、选择题（每题 1 分）：从每题给出的四个答案中，在 ABCD 中选择一个正确答案，填入【     】。

1、 采用 C++面向对象程序设计来设计软件有许多优点，但是不包括【     】。

- A. 有利于信息隐藏和数据安全
- B. 有利于软件重用，提高软件开发效率
- C. 提高模块独立性，有利于软件维护
- D. 降低对于软件开发人员的要求

2、 以下定义的 C++实数中，不正确的是【     】

- A. float a=3.54e2
- B. float a=1.0e2
- C. float a=.54e2
- D. float a=3.54e2.0

3、 在字符串中使用以下字符（字符组合）时，不需要采用转义字符的是【     】。

- A. \
- B. '
- C. \
- D. "

4、 语句 for(int i=1;i++<4;); 虽然正确，但可读性差，可以采用的等价语句是【     】。

- A. for(int i=2;i<5;i++);
- B. for(int i=1;i<4;i++);
- C. for(int i=1;i<=4;i++);
- D. for(int i=1;i<3;i++);

5、 若已定义 char \*p="abcd", \*q="cdef"; 以下语句中会出现运行错误的是【     】。

- A. cout<<p;
- B. p="efgh";
- C. p=q;
- D. \*p='P';

6、 已声明原型 void func(int arr[]); 在函数 func 中要知道数组 arr 的大小,可使用【     】。

- A. sizeof(arr)/sizeof(int)
- B. sizeof(arr[])/sizeof(int)
- C. sizeof(int arr[])/sizeof(int)
- D. 其他方法，以上方法都不正确

7、 为了判断两个字符串 s1 和 s2 是否相等，应当使用【     】。

- A. if ( s1==s2 )
- B. if ( s1=s2 )
- C. if (!strcmp(s1, s2))
- D. if ( strcmp(s1, s2))

8、 若已定义 int k=1; 然后不断地执行 k++操作，k 值变化的趋势将是【     】。

- A. k 值不断增加到最大值，显示溢出错误后停止
- B. 从 1 增加到最大值，再从-1 减少到负数最小值，周而复始
- C. 从 1 增加到最大值，再从负数最小值增加到-1，周而复始
- D. 从 1 增加到最大值，再减少到 1，周而复始；

9、 设有定义语句

int A[5]={2,4,6,8,5}, \*p=A, i;

要求依次输出 A 数组 5 个元素的值，不能完成此操作的语句是【     】。

- A. for(i=0;i<5;i++) cout<<\*(p++);
- B. for(i=0;i<5;i++) cout<<\*(p+i);
- C. for(i=0;i<5;i++) cout<<p[i];
- D. for(i=0;i<5;i++) cout<< (\*p)++;

10、 下列函数中，不是类的成员函数的是【     】。

- A. 友元函数
- B. 析构函数

- C. 将数据转换为类的对象的转换函数      D. 拷贝构造函数
- 11、以下对于构造函数说法中，正确的是【      】。
- A. 构造函数通常是自动调用的      B. 构造函数返回值可以是 void 型
- C. 无参数构造函数可以不止一个      D. 一个类中只能定义一个构造函数
- 12、类的对象可以访问的类的成员必须是【      】类型。
- A. auto      B. private      C. protected      D. public
- 13、以下对于析构函数说法中，正确的是【      】。
- A. 每个类必须自己定义一个析构函数      B. 析构函数可以有形式参数
- C. 一个类中只能定义一个析构函数      D. 析构函数需要在程序中调用
- 14、要在 Clock 类外定义一个构造函数，类中包含一个整型的数据成员 m。以下 4 种函数头表示中，正确的是【      】。
- A. void Clock::Clock (int m)      B. Clock::Clock (int m)
- C. int Clock::Clock (int m)      D. Clock (int m=0)
- 15、如果已经定义了 Point 类的静态数据成员：static int k; 对这个数据成员初始化的正确表示是【      】。
- A. int Point::k=0;      B. static int k=0;
- C. static int Point::k=0;      D. int static Point:: k=0;
- 16、友元函数和类的成员函数相同的地方是【      】。
- A. 都可以通过类的对象来调用      B. 都可以访问类的私有成员
- C. 都可以通过类的对象指针来调用      D. 都有 this 指针
- 17、关于 new 运算符的下列描述中，错误的是【      】。
- A. 可以用来动态地创建对象或变量
- B. 使用它来创建对象时要调用构造函数
- C. 使用它创建对象时可以指定初始值
- D. 使用它创建对象数组时可以指定初始值
- 18、关于派生类 newClass 的以下声明中，正确的是【      】。
- A. class newClass: public class Base{....};
- B. class newClass: public Base01, Base02{....};
- C. class newClass: public Base01, public Base02{....}
- D. class newClass: public class Base01, public Base02{....};;
- 19、以下操作中不可以进行的是【      】。
- A. 将派生类对象赋值给基类对象
- B. 将基类对象赋值给派生类对象
- C. 用基类对象的地址初始化基类的指针
- D. 用派生类对象的地址初始化基类的指针
- 20、以下说法中错误的是【      】。
- A. 不可以定义抽象类的对象
- B. 不可以定义抽象类对象的指针
- C. 抽象类中必须定义纯虚函数
- D. 纯虚函数返回值可以不是 void 类型



### 三、分析题（每题 4 分）：按要求回答问题

1、 以下程序的输出结果是\_\_\_\_\_

程序的功能是\_\_\_\_\_

```
#include <iostream>
using namespace std;
void main()
{   int n=0;
    for (int i=1;i<20;i++)
    {
        if ((i % 3))
            continue;
        n+=i;
    }
    cout<<"n="<<n<<endl;
}
```

2、 以下程序的输出结果是\_\_\_\_\_

```
#include <iostream>
using namespace std;
void fun(int *p,int n)
{for(int i=0;i<3;i++)
  {*(p+i)=p[i]*i;
   n++;
  }
}
void main()
{   int a[3]={4,5,6},n=3;
    fun(a,n);
    cout<<"n="<<n<<endl;
    cout<<"result="<<(a[0]+a[1]+a[2])/n<<endl;
}
```

3、 以下程序的输出结果是\_\_\_\_\_

```
#include <iostream>
using namespace std;
class A{
public:
    A(){}
    virtual void func() {cout<<"Destructor A"<<endl;}
    ~A() {func();}
};
class B:public A{
```

```

public:
B(){}
void func(){cout<<"Destructor B"<<endl;}
~B() {func();}
};
void main( )
{ B b;
  A *a=&b;
}

```

- 4、 weight 类对于前置++运算符作了重载，只是返回值是 weight 类对象而不是引用。

以下程序运行结果为\_\_\_\_\_

\_\_\_\_\_（注：程序可以正常运行）

```

#include <iostream>
using namespace std;
class weight
{public:
    weight(int v=0):value(v) {}
    weight operator ++();
    void print()
    { cout<<value<<endl;
    }
private:
    int value;
};
weight weight::operator ++()
{ value++;
  return *this;
}
void main()
{ weight s1(5);
  (++s1).print();
  s1.print();
}

```

- 5、 以下程序的输出结果为： \_\_\_\_\_  
 \_\_\_\_\_

```

#include <iostream>
using namespace std;
class Base
{public:
    Base(){num+=2; }
    void print ()const
    {cout<<"num="<<num<<endl;}
}

```

```

void print ()
{cout<<"num+2="<<num+2<<endl;}
    static int num;
};
int Base::num=0;
void main()
{Base  B[3];
  B[0].print();
cout<<++B[1].num<<endl;
}

```

#### 四、改错题（每题 4 分）：按要求答题和改错

- 1、 以下程序显示所有可显示的 ASCII 字符，其码值从 32 到 127。但循环不能结束。

```

#include <iostream>
using namespace std;
void main()
{ int i=1;
  char ch=32;
  while(ch+129)
  { cout<<(int)ch<<" "<<ch<<" ";           //显示码值和字符
    ch++; i++;
    if(i==10)                                //一行显示 10 个字符
    {cout<<endl; i=1;}
  }
}

```

导致错误结果的语句成份是 \_\_\_\_\_，应该修改为\_\_\_\_\_。

- 2、 以下程序用函数 sum\_array 求一维数组和二维数组元素的和。程序出现一处编译错误，请找出这个错误并改正。

```

#include <iostream>
using namespace std;
long sum_array(int array[], int elements)
{ long sum = 0;
  for (int i = 0; i < elements; i++)
    sum += array[i];
  return(sum);
}
void main(void)
{ int a[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
  int b[2][10] = {{1, 2, 3, 4, 5, 6, 7, 8, 9, 10},
                  {11, 12, 13, 14, 15, 16, 17, 18, 19, 20}};
  cout<<"Sum of first array elements is "<<sum_array(a, 10)<<endl;
  cout<<"Sum of second array elements is "<<sum_array(b, 20)<<endl;
}

```

导致错误结果的语句成份是 \_\_\_\_\_，应该修改为\_\_\_\_\_。

- 3、 指出下面类定义程序段中的错误（编译错误），说明出错原因，并加以改正。

```
class Shape{
    private:
        static int x=25;
        int y;
    public:
        Shape(int Y=0){ y=Y;}
        int getX(){return x;}
        int getY(){return y;}
};
```

错误位置和原因：\_\_\_\_\_

修改：\_\_\_\_\_

- 4、 指出下面程序段中的错误（编译错误），说明出错原因，并加以改正。

```
#include <iostream>
using namespace std;
class A{
    int a,b;
    friend int add_member();
public:
    A(int aa, int bb) {a=aa;b=bb;}
};
int add_member(A &obj)
{ return obj.a+obj.b;}
void main()
{A x(2,3);
    cout<<add_member(x)<<endl;
}
```

位置及原因：\_\_\_\_\_

修改为：\_\_\_\_\_

- 5、 以下程序希望运行时显示 p=20，但编译时出现错误。说明系统给出的错误信息将是什么（用中文写就可以），并加以改正。注意：不修改 main 函数中的语句。

```
#include <iostream.h>
class Test{
    public:
        Test(int w, int x){p=w; q=x;}
        void print() {cout<<"p="<<p<<endl;}
    private:
        int p, q;
};
void main()
{Test obj01;
    obj01.print();
}
```

错误的语句成份及原因：\_\_\_\_\_

改正\_\_\_\_\_

## 五、完成程序题(每空 1 分，共 20 分)

- 1、 下面程序中的函数 `string_uppercase` 返回一个全部是大写字母的字符串，请填空。其中的库函数 `toupper` 完成将小写字母到大写的转换，但是返回值是 `int` 型。

```
#include <iostream>
using namespace std;
char *string_uppercase(char *string)
{ char *new_address,*temp;
  new_address = new _____;
  temp=new_address;
  while (_____)
    *new_address++=(char)toupper(*string++);
  *new_address=_____;
  return(_____);
}
void main(void)
{ char *title = "C++ is a better C.";
  char *string;
  string = string_uppercase(title);
  cout<<string<<endl;
  delete _____;
}
```

- 2、 完成下面类的声明和成员函数的定义。由于数组 `name` 足够大，初始化姓名时不必考虑数组长度不够的问题。并完成对于类的定义的测试。

```
#include <iostream>
using namespace std;
class classA
{public:
  classA(char *p, char q)           //构造函数定义
  {_____};                       //初始化姓名(name)
  sex=_____;
  *sex=_____;                     //初始化性别
}
classA(classA &c)                   //拷贝构造函数定义
{_____};                           //初始化姓名(name)
  sex=_____;
  *sex=_____;                     //初始化性别
}
~classA()
{_____};
}
void print()
{cout<<"name="<<name<<endl;
```

```

        cout<<"sex="<<*sex<<endl;
    }
private:
    char name[30];
    char *sex;
};
void main()
{classA  A1("WangYi",'M');
  classA  A2=A1;
  _____;           //显示对象 A2 的信息
}

```

- 3、 以下程序用冒泡排序法对于职工信息按工资递增进行排序，请填空，使得程序完整和正确。

```

#include <iostream>
using namespace std;
struct Employee
{ char name[20];
  float salary;
};
Employee allone[6]={{"zhang", 3390.0}, {"wang", 4490.0},
                    {"zhou", 3110.0}, {"meng", 6230.0},
                    {"yang", 4000.0}, {"chen", 5110.0}};
void bubble_sort(Employee all[],int n)
{ Employee temp;
  for(int _____; i<n; i++)           //排序
  { for(int _____; j<n-i; j++)       //一轮比较
    { if(_____ )                       //比较工资成员
      { temp=_____ ;                   //结构变量的交换
        all[j]=_____ ;
        _____=temp;
      }
    }
  }
}
void main()
{ bubble_sort(_____);
  for(int k=0; k<6; k++) //输出
    cout <<allone[k].name<<" " <<allone[k].salary<<endl;
}

```

## 2005C++ 北邮

### 一、填空题

- 1、 继承 多态
- 2、 属性 行为
- 3、 7.0/2
- 4、 iomanip
- 5、 void setname(char \*s)
- 6、 27
- 7、 true
- 8、 private
- 9、 int fun(int& , int &)
- 10、 构造函数 拷贝构造函数
- 11、 拷贝构造函数
- 12、 抽象类
- 13、 static extern
- 14、 无限
- 15、 公有继承 基类公有成员

### 二、选择题

- 1、 D
- 2、 D
- 3、 B
- 4、 A
- 5、 D
- 6、 A
- 7、 C
- 8、 C
- 9、 D
- 10、 A
- 11、 A
- 12、 D
- 13、 C
- 14、 B
- 15、 A
- 16、 B
- 17、 D
- 18、 C
- 19、 B
- 20、 B

### 三、分析题

- 1、 53 计算  $3+6+9+12+15+18$  的值
- 2、 3 5
- 3、 Destructor B  
Destructor A
- 4、 7 6

5、num+2=8 7

#### 四、改错

1、错误位置：while(ch+129)

修改：ch>0

2、错误位置：sum\_array(b,20)

修改：sum\_array(b[0],20);

3、错误位置：static int x=25; 成员变量不能在声明的时候初始化

修改：static int x; int Shape::x =25;

4、错误位置：friend int add\_member(A&)

修改：添加前项声明 class A; 和 int add\_member(A&)

5、错误位置：Test(int w,int x){p=w; q=x;}, 没有适合的构造函数

修改：Test(int w=0,int x=0){p=w; q=x;}

#### 五、程序填空

1、 new\_address = new char[strlen(string)+1]

while(\*string!='\0')

\*new\_address='\0'

return temp;

delete string;

2、 strcpy(name,p);

sex =new char

\*sex = q

strcpy(name,c.name)

sex = new char;

\*sex= \*c.sex

delete sex;

A2.print();

3、 i=1

j=0

all[j]>all[j+1]

temp = a[j];

a[j] = a[j+1];

a[j+1]=temp;

bubble\_sort(allone, 6);



# 北京邮电大学 2005——2006 学年第二学期

## 《C++程序设计》期末考试试题

### 一：填空题（30 分）

1. C++的逻辑运算符是\_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_, 分别完成\_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_运算。

2. 算法和程序的三种基本控制结构是\_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_。

3. C++源文件扩展名是\_\_\_\_\_和\_\_\_\_\_, 类声明放在\_\_\_\_\_文件中, 类实现在\_\_\_\_\_文件中。

4. 作用域是\_\_\_\_\_。C++的作用域分为\_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_。

5. \_\_\_\_\_运算符用于计算某种类型的对象在内存中所占的字节数, 其语法形式为\_\_\_\_\_。

6. This 指针是指向\_\_\_\_\_对象的。

7. 派生类对其基类构造函数的调用和对内嵌子对象的初始化, 应在\_\_\_\_\_处完成。

8. 带有纯虚函数的类是\_\_\_\_\_。

9. 类的继承中, 有三种继承方式\_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_。

10. 运算符重载有两种形式: 可以重载为类的\_\_\_\_\_函数或类的\_\_\_\_\_函数

11. 执行语句 `char* p;`将在内存分配\_\_\_\_\_字节的存储空间。

12. 执行语句

```
struct person
```

```
{int num;
```

```
union
```

```
{ int class;
```

```
    char position[10];
```

```
}
```

```
}student1;
```

 将在内存分配\_\_\_\_\_字节的存储空间。

### 二：单选题（20 分）

1、C++对 C 语言作了很多改进, 下列描述中 ( ) 使得 C 语言发生了质

变，从面向过程变成了面向对象。

- A. 增加了一些新的运算符；
- B. 允许函数重载，并允许设置缺省参数；
- C. 规定函数说明必须用原型；
- D. 引进了类和对象的概念；

2、考虑函数原型 `void test(int a, int b=7, char *s="*")`, 下面的函数调用中, 属于不合法调用的是 ( )

- A. `test(5);`
- B. `test(5,8);`
- C. `test(6,"#");`
- D. `test(0,0,"*");`

3、下面有关重载函数的说法中正确的是 ( )

- A. 重载函数必须具有不同的返回值类型；
- B. 重载函数形参个数必须不同；
- C. 重载函数必须有不同的形参列表
- D. 重载函数名可以不同；

4、下列关于构造函数的描述中，错误的是 ( )

- A. 构造函数可以设置默认参数；
- B. 构造函数在定义类对象时自动执行
- C. 构造函数可以是内联函数；
- D. 构造函数不可以重载

5、下面描述中，表达错误的是 ( )

- A. 公有继承时基类中的 `public` 成员在派生类中仍是 `public` 的
- B. 公有继承是基类中的 `private` 成员在派生类中仍是 `private` 的
- C. 公有继承时基类中的 `protected` 成员在派生类中仍是 `protected` 的
- D. 私有继承时基类中的 `public` 成员在派生类中是 `private` 的

6、友元的作用是 ( )。

- A. 提高程序的运用效率；
- B. 加强类的封装性；
- C. 实现数据的隐藏性；
- D. 增加成员函数的种类；

7、关于 `new` 运算符的下列描述中，( ) 是错误的。

- A. 它可以用来动态创建对象和对象数组；
- B. 使用它创建的对象或对象数组可以使用运算符 `delete` 删除；
- C. 使用它创建对象时要调用构造函数；
- D. 使用它创建对象数组时必须指定初始值；

8、派生类的对象对它的基类成员中 ( ) 是可以访问的。

- A. 公有继承的公有成员;
- B. 公有继承的私有成员;
- C. 公有继承的保护成员;
- D. 私有继承的公有成员;

9、设置虚基类的目的是 ( )。

- A. 简化程序;
- B. 消除二义性;
- C. 提高运行效率;
- D. 减少目标代码;

10、在进行完任何 C++流的操作后，都可以用 C++流的有关成员函数检测流的状态；其中只能用于检测输入流状态的操作函数名称是 ( )

- A. fail    B. eof    C. bad    D. good

11、分析下面的程序，运行结果为\_\_\_\_\_

```
main(){
    int a[]={1,2,3,4,5,6,7,8,9};
    int *p=a;
    cout<<*p++;
    cout<<*(p+1);
}
```

- A. 1 3    B. 2 2    C. 1 2    D. 2 3

12、下面程序的输出结果是\_\_\_\_\_

```
void main(){
    int a=1,b=10;
    int &r = a;
    int *p = &b;
    *p=r++;
    cout<<a<<","<<b<<endl;
}
```

- A. 1,10    B. 2,10    C. 1,2    D. 2,1

13 有以下程序段

```
main ( ) {
    int a=5, *b, **c;
    c=&b;
    b=&a;
    ...
}
```

程序在执行了 c=&b;b=&a;语句后，表达式:\*\*c 的值是 \_\_\_\_\_

- A. 变量 a 的地址    B. 变量 b 中的值
- C. 变量 a 中的值    D. 变量 b 的地址

14、下面函数中能够体现面向对象的多态性的是\_\_\_\_\_

- A. 构造函数    B. 虚函数    C. 析构函数    D. 友元函数

15、构造一个类 A，在该类中的将“+”操作符重载为友元函数，则下面语法正确的是\_\_\_\_\_ A C 都正确

- A. `friend A operator +=(A&, A&)`  
B. `friend A& operator +=(A ,A)`  
C. `friend A& operator +=(A&, A&)`  
D. `A& operator +=(A&)`

16、下面哪一个变量占用了内存空间\_\_\_\_\_

- A. `enum COLOR{RED ,BLUE, GREEN};` 中的 COLOR  
B. `int a; int &r=a;` 中的变量 r  
C. `const int *p;` 中的 p  
D. `struct Student`

```
{  
    int NO;  
    char name[20];  
}; 中的 Student;
```

17、下列哪些函数有返回值\_\_\_\_\_

- A. 转换运算符函数    B. 构造函数  
C. 析构函数    D. 赋值运算符函数

18、如果让类的所有成员在类的范围内共享某一个函数，则声明该函数为\_\_\_\_\_

- A. static    B. friend    C. virtual    D. const

19、构造一个类时，类中有些函数系统缺省会进行构造，那么下面哪一类系统不会产生默认的函数 \_\_\_\_\_

- A. 构造函数    B. 友元函数  
C. 拷贝构造函数    D. 赋值运算符

20、在类的继承过程中，哪些函数能够被继承\_\_\_\_\_

- A. 虚函数    B. 构造函数  
C. 析构函数    D. 友元函数

### 三：简答题（20 分）

1、分析程序，给出输出结果。

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

```

class Base
{
public:
    Base(int i=3);
    virtual void Print() const;
    int GetValue() const;
private:
    int MyValue;
};

Base::Base(int v):MyValue(v){}
void Base::Print()const
{cout<<"Base Class Print: "<<GetValue()<<endl;
}
int Base::GetValue()const
{return MyValue;
}

class Derived:public Base
{
public:
    Derived(int i=12);
    virtual void Print() const;
    int GetValue() const;
private:
    int MyValue;
};

Derived::Derived(int v):MyValue(v){}
void Derived::Print()const
{cout<<"Derived Class Print: "<<GetValue()<<endl;
}
int Derived::GetValue()const
{return MyValue;
}

void main()
{
    Base B(1);
    Derived D;
    Base *pb1 = new Base;
    Base *pb2 = new Derived(5);
    Derived *pd1=new Derived(10);

    B.Print();
    D.Print();
}

```

```

        pb1->Print();
        pb2->Print();
        pb2=pd1;
        pb2->Print();
    }

```

2、下面的函数将字符串 s 中字符倒序，使用指针 t 指向字符串最后一个字符，然后首尾字符相互交换的方法实现，比如 s = “Hello”，经过该函数后 s= “olleH”，将下面的算法补充完全。

```

void reverse(char *s)
{
    char c;
    char *t;
    for (t = s + strlen(s) - 1; _____;t--,s++)
    {
        ch = *s ;
        _____;           //字符首尾字符交换
        _____;
    }
}

```

3、输入 elephant 后，下列程序的输出结果是：

```

#include <iostream.h>
void main()
{   int i=0, k;
    static int num[5];
    char alpha[ ]={'a', 'e', 'i', 'o', 'u'}, in[80];
    cin>>in;
    while(in[i])
    {
        for(k=0; k<5; k++)
            if(in[i] == alpha[k])
            {
                num[k]++;
                break;
            }
        i++;
    }
    for(k=0; k<5; k++)
        if(num[k])

```

```
        cout<<alpha[k]<<num[k];  
        cout<<endl;  
    }
```

4、把文本文件 input.txt 中的非空格字符输出到屏幕显示。

四、(15 分)

```
#include <string.h>  
class A{  
public:  
    A(char *pName="noname")  
    {  
        name=new char[strlen(pName)+1];  
        strcpy(name,pName);  
    }  
protected:  
    char *name;  
};  
void main()  
{  
    A a;  
    A b(a);  
}
```

1) 程序运行中会产生内存泄漏，为什么？

2) 请添加一个析构函数，避免 1) 中的问题

3) main()函数中调用了拷贝构造函数，当只添加析构函数时，程序会出现什么问题？添加一个拷贝构造函数，避免该问题。

## 五、(15 分)

要求：

(1) 创建一个基类X, 包含一个int型数据成员，其构造函数含有单个int参数，在构造函数内完成其数据成员的初始化。其成员函数f()返回为void型，无参数，在该函数内将其数据成员设置为0, 并输出显示。

(2) 从X派生出Y和Z，各含一个int型数据成员，为Y和Z各创建一个带int参数的构造函数，在构造函数内完成其数据成员的初始化。

(3) 通过多重继承从Y和Z中派生出A, 它也有一个int型数据成员，为A创建一个带参数的构造函数，在构造函数内完成其数据成员的初始化。

(4) 生成一个类A的对象并通过对象调用f()。以明显无二义性的方式解决这个问题。

程序框架如下：

```
#include _____(1)//包含的头文件
using namespace std;
```

```
class X
```

```
{
```

```
public:
```

```
    //构造函数
```

```
    _____(2)
```



```

        {
            _____(3)
        }
        void f(){
            _____(4)
            _____(5)
        }
private:
    int m_x;
};
class Y: _____(6)
{
public:
    //构造函数
    _____(7)
    {
        _____(8)
    }
private:
    int m_y;
};
class Z: _____(9)
{
public:
    //构造函数
    _____(10)
    {
        _____(11)
    }

private:
    int m_z;
};
class A: _____(12)
{
public:
    //构造函数
    _____(13)
    {
        _____(14)
    }
}

```

```
private:
    int m_a;
};

void main()
{
    A aobj(1,2,3,4);
    aobj.f();
}
```

**对外经济贸易大学 2000-2001 学年第一学期  
C 语言程序设计期中测验试卷**

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

| 一 | 二 | 三 | 四 | 五 | 六 | 总分 |
|---|---|---|---|---|---|----|
|   |   |   |   |   |   |    |

**一、选择题（每小题 2 分，共 40 分）**

- 1 组成 C 语言程序的是【    】  
(A) 子程序 (B) 过程  
(C) 函数 (D) 主程序和子程序
- 2 设  $a=2, b=3$ , 执行表达式  $c = b * a - 1$ ; 变量  $c$  的值是【    】  
(A) 5 (B) 3 (C) 2 (D) 4
- 3 若  $a$  是 `int` 型变量, 则表达式  $(a=4*5, a*2), a+6$  的值为【    】  
(A) 20 (B) 40 (C) 46 (D) 26
- 4 C 语言的运算符以运算对象的数目可以分为【    】  
(A) 单目算符一种 (B) 单目算符和双目算符  
(C) 单目、双目和三目算符 (D) 单目、双目、三目和四目算符
5. 运算符是有优先级的, 在 C 语言中下列哪一个叙述是正确的?【    】  
A. 逻辑运算符高于算术运算符, 算术运算符高于关系运算符  
B. 算术运算符高于关系运算符, 关系运算符高于逻辑运算符  
C. 算术运算符高于逻辑运算符, 逻辑运算符高于关系运算符  
D. 关系运算符高于逻辑运算符, 逻辑运算符高于算术运算符
6. C 语言并不是非常严格的算法语言, 以下关于 C 语言不严格性的叙述中, 不正确的是:【    】  
A. 大写字母和小写字母的意义相同  
B. 有些不同类型的变量可以在一个表达式中运算  
C. 表达式左边变量和右边的值可以是不同类型的  
D. 同一个运算符在不同的场合可以有不同的含义
7. 以下关于 C 语言字符数组的说法中, 不正确的是:【    】  
A. 在 C 语言中可用字符数组存放字符串  
B. 可以用关系运算符对字符数组进行比较  
C. 字符数组中的字符串可以整体输入、输出  
D. 字符数组中的字符串可以按单个字符输入、输出
8. 在 C 语言中, 数字 029 是一个【    】  
A. 八进制数 B. 十六进制数 C. 十进制数 D. 非法数
9. 选出年龄在 30 到 40 (包括 30, 不包括 40) 之间的人, 其 C 语言表达式为:【    】  
A.  $(\text{年龄} \geq 30) \text{ and } (\text{年龄} < 40)$   
B.  $(40 > \text{年龄} \geq 30)$   
C.  $(\text{年龄} \geq 30) \& (\text{年龄} < 40)$   
D.  $(\text{年龄} \geq 30) \&\& (\text{年龄} < 40)$
10. 若已定义  $a, b$  为 `int` 变量, 用 `scanf("%d%d", &a, &b);` 语句输入  $a, b$  的值时, 不可以作为输

入分隔符的为：【 】

A. ,      B. 空格      C. 回车      D. Tab

11.若有 int i, 则表达式 i=1,i+1,i++ 的值为: 【 】

A.1      B.2      C.3      D.4

12. 表达式 3&&6&&9 的值为: 【 】

A.3      B.6      C.9      D.1

13.执行以下程序的输出是? 【 】

```
main()
{int x=8;
 float y=-2.6;
 printf("%f\n",x+y);
}
```

A.错误信息    B.6      C.5      D.5.400000

14.执行以下程序的输出是? 【 】

```
main()
{int x,a=1,b=2,c=0;
 x=-a||b&&++c;
 printf("%d\n",x);
}
```

A.1      B.0      C.-1      D.非零值

15.执行以下程序的输出是? 【 】

```
main()
{int x=1,y=2,z;
 z=x+++y;
 printf("%d%d%d\n",x.y.z);
}
```

A.223      B.123      C.224      D.134

16.执行以下程序的输出是? 【 】

```
main()
{int a=10,b=0;
 if (a=12) {a=a+1;b=b+1;}
 else {a=a+4;b=b+4;}
 printf("%d,%d\n",a,b);
}
```

A.13;1      B.14;4      C.11;1      D.10;0

17. 若在 C 语言程序中使用以下的二维数组定义语句并赋以部分初值，其中不正确的定义只有：【 】

A.int a[2][3]={ {4},{5,6}}

B.int a[2][3]={ {4},{5},{6}}

C. int a[2][3]={4,5,6}

D. int a[2][3]={ {4},5,6}

18 若有以下定义和语句，则输出结果是【 】

```
char s[12]="a book!";
printf("%d", strlen(s));
```

(A) 12

(B) 8

(C) 7

(D) 6

19 以下程序的运行结果是【 】

```

main()
{int i=5;
 printf("%d\n",sub(i));
}
sub(int n)
{int a;
 if (n==1) return 1;
 a=n+sub(n-1);
 return(a);
}

```

- (A) 15 (B) 21 (C) 1 (D) 11
- 20 以下程序的输出是 【     】

```

try(int a)
{ static int x = 3;
  x = x+1; return x+a; }
main()
{ int a, x=2;
  for(a = 0; a <= 1; a++) x=try(x);
  printf("%d\n",x); }

```

- (A) 5 (B) 7 (C) 10 (D) 11
- 二、写出程序运行结果（共 20 分）

1 写出程序运行结果（6 分）

```

#include<stdio.h>
main()
{ int k = 0; char c = 'A';
  do {
    switch(c++)
    { case 'A': k++; break
      case 'B': k--;
      case 'C': k+=2; break;
      case 'D': k%=2; continue;
      case 'E': k=k*2; break;
      default: k=(c?'1':0); }
    k++;
    printf("%c k = %d\n",c-1, k);
  } while (c<'F');
}

```

运行结果:

2 下列程序的两个 printf 输出结果按其先后顺序是 【     】，【     】（4 分）

3 以下程序段的两个 printf 输出结果分别是 【     】 【     】（4 分）

4 写出运行结果（6 分）

```

main()

```

```
q++; }
```

输出结果【

】

```
}
```

### 三 程序选择填空（10 分）

1

### 四 程序填空及编程（30 分）

1 以下的程序将输入的十进制整数  $n$  转换为  $b$  进制数( $b \leq 8$ ), 并将转换结果输出。转换的方法是“除  $b$  取余”。请填空。(15 分)

```
void transfer(int m,int k)
{int aa[20],i;
 for (i=0;m;i++)
  {aa[i]=【    】; m/=k;}
  【    】 printf("%d", 【    】);
}
main()
{int b,n;
 printf("\nPlease enter a number and a base:\n");
 scanf("%d%d", 【    】);
  【    】;
}
```

2 请编写函数 `void fun(int *a, int y, int n)`, 它的功能是: 把  $y$  值插入到  $a$  数组中,  $a$  数组中的数由主函数按由小到大的次序从键盘读入,  $y$  是要插入的数, 其值也从键盘读入。  $n$  是  $a$  中数字的个数( $n < M$ )。要求插入后, 数组中的数仍然有序。

```
#define M 50
```

(15 分)

```
void fun (int *a, int y, int n)
{
}

}
```

/\*答案开始 \*/

```
⋮
```

```
⋮
```

/\*答案结束 \*/

一. 选择

1—5: CBCCB      6—10: ABADA    11—15: ADDAA    16—20: ABCAD

二

1. 运行结果:

A    k=2

B    k=4

C    k=7

E    k=15

四

1

m%k

for (;i--)

aa[i]

b,n

transfer(b,n)

对外经济贸易大学 2000-2001 学年第一学期

C 语言程序设计期末考试试卷

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

| 一 | 二 | 三 | 四 | 五 | 六 | 总分 |
|---|---|---|---|---|---|----|
|   |   |   |   |   |   |    |

一、选择题（每小题 2 分，共 40 分）

1 组成 C 语言程序的是【 】

- (A) 子程序 (B) 过程  
(C) 函数 (D) 主程序和子程序

2 若有以下定义和语句，且  $0 \leq i < 10$ ，则对数组元素的错误引用是【 】

int a[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 0};

int \*p, i; p=a;

- (A) \*(a+i) (B) a[p-a] (C) p+i (D) \*(&a[i])

3 若有下列定义，则对 a 数组元素地址的正确引用是【 】

int a[5], \*p=a;

- (A) &a[5] (B) p+2 (C) a++ (D) &a

4 若有以下定义和语句，则输出结果是【 】

char s[12]="a book!";

printf("%d", strlen(s));

- (A) 12 (B) 8 (C) 7 (D) 6

5 以下程序的结果是【 】

main()

{ unsigned int a=3, b=10; printf("%d\n", a << 2 | b >> 1); }

- (A) 13 (B) 12 (C) 5 (D) 1

6 以下程序的运行结果为【 】

#define PT 3.5

#define S(x) PT\*x\*x

main()

{ int a=1, b=2; printf("%4.1f\n", S(a+b)); }

- (A) 7.5 (B) 10.5 (C) 14.0 (D) 31.5

7 C 语言中，文件由【 】

- (A) 记录组成 (B) 由数据行组成 (C) 由数据块组成 (D) 由字符序列组成

8 typedef long int BIGGY; 的作用是【 】

- (A) 建立了一种新的数据类型 (B) 定义了一个整型变量  
(C) 定义了一个长整型变量 (D) 说明了一个新的数据类型标识符

9 若有以下的说明、定义和语句，则值为 31 的表达式是【 】

struct wc

{ int a;

int \*b;

} \*p;

int x0[] = {11, 12}, x1[] = {31, 32};



static struct wc x[2]= { 100, x0, 300, x1 } ;

p=x;

(A) \*p -> b      (B) \*(++p) -> b      (C) \*(p++) -> b      (D) (++p) -> a

10 与 int \*p[4]; 定义等价的是 【    】

(A) int p[4];      (B) int \* p ;      (C) int \* (p[4]);      (D) int (\*p)[4];

11 若有定义: char \*p1, \*p2, \*p3, \*p4, ch; 则不能正确赋值的程序语句为 【    】

(A) p1=&ch; scanf("%c", p1);      (B) p2=getchar();  
(C) p3=(char\*)malloc(1); scanf("%c", p3);      (D) p4=&ch; \*p4=getchar();

12-14 下面程序的输出分别是:

main()

{ char \*s2="sceg";

s2++;

printf("%c\n", \*s2);      /\* 【    】 \*/

printf("%c\n", \*(s2+1));      /\* 【    】 \*/

printf("%c\n", \*s2+1); }      /\* 【    】 \*/

12-14、 A) a      B) b      C) c      D) d      E) e

15 设 a=2, b=3, 执行表达式 c = b\* = a-1; 变量 c 的值是 【    】

(A) 5      (B) 3      (C) 2      (D) 4

16-17 执行下面程序的输出结果分别是 【    】、【    】

void change ( int x, int m )

{ static char ch[ ] = "0123456789";

int I = 0, r ;    char b[80];

while ( x )

{ r = x % m; x /= m; b[I++] = ch[r]; }

b[I] = '\0';

printf("%s\n", b); }

main( )

{ change( 42, 2 ); change (42, 8 ); }

16) (A) 010101      (B) 101010      (C) 10101      (D) 01010

17) (A) 52      (B) 25      (C) 40      (D) 24

18 以下程序的输出是 【    】

#include<stdio.h>

main()

{ int i, j, k, a=3, b=2;

i = (--a == b++) ? --a : ++b;

j = a++; k=b;

printf("i=%d, j=%d, k=%d\n", i, j, k);

}

(A) i=2, j=1, k=3

(B) i=1, j=1, k=2

(C) i=4, j=2, k=4

(D) i=1, j=1, k=3

19 以下程序的输出是 【    】

#include<stdio.h>

main( )

{static int a[4][3] = {1,2,3,4,5,6,7,8,9,10,11,12};

```
int *x[4] = { a[0], a[1], a[2], a[3] };
int j, **p;
p = x; p++;
for (j=2; j>=0; j--)
    printf("%3d", *(*(p)+j)); }
```

19) (A) 3 2 1 (B) 1 2 3 (C) 6 5 4 (D) 4 5 6

20 以下程序的输出是 【 】

```
try(int a)
{ static int x = 3;
  x = x+1;
  return x+a; }
main()
{ int a, x=2;
  for(a = 0; a <= 1; a++) x=try(x);
  printf("%d\n",x); }
```

A) 5 B) 7 C) 10 D) 11

## 二、写出程序运行结果（共 20 分）

1 写出程序运行结果（6 分）

```
#include<stdio.h>
main()
{ int k = 0; char c = 'A';
  do {
    switch(c++)
    { case 'A': k++; break;
      case 'B': k--;
      case 'C': k+=2; break;
      case 'D': k%=2; continue;
      case 'E': k=k*2; break;
      default: k=(c?1:0); }
    k++;
    printf("%c k = %d\n",c-1, k);
  } while (c<'F');
}
```

2 以下程序段的两个 printf 输出结果分别是 【 】，【 】（4 分）

```
int a=3, b=-1;
printf("%u\n",a >> 1 ^ b << 1);
printf("%d %o\n", b, b);
```

3 以下程序段的两个 printf 输出结果分别是 【 】，【 】（4 分）

```
union { int a[3];
        long b;
        char c[4]; } s;
s.a[0] = 0x39; s.a[1] = 0x38;
printf("%lx\n", s.b);
```

```
printf("%d\n", s.c[1]);
```

4 已知数组 a 的首地址是 65482，写出运行结果（10 分）

```
main()
```

```
{ static int a[3][2]={1, 3, 5, 7, 9, 11};
```

```
int *p, (*k)[2];
```

```
p=a[0]; k=a;
```

```
printf("%u\n", a);
```

```
printf("%u\n", *((a)+1));
```

```
printf("%u\n", p+2);
```

```
printf("%u\n", *(p+2));
```

```
printf("%u\n", *p++);
```

```
printf("%u\n", (*p)+1);
```

```
printf("%u\n", *(k+2));
```

```
printf("%u\n", *((k+2)));
```

```
}
```

### 三 程序填空（20 分）

1、（10 分）函数 index(): 从指针变量 p 所指的字符串中找指针变量 q 所指的字符串的位置，若找不到则输出 0，例如 当 string1 为 abcdef, string2 为 cde, 则输出 3, string2 为 cdf, 则输出 0.

```
main()
```

```
{ char 【1】
```

```
printf("Input string1 and string2");
```

```
【2】
```

```
printf("%d\n", 【3】);
```

```
}
```

```
index(char *p, char *q)
```

```
{ char *pp, *qq;
```

```
int i;
```

```
for(i=1, pp=p; *pp!='\0'; i++, pp++)
```

```
{ for(【4】 *qq==*pp && *qq!='\0'; pp++, qq++);
```

```
if(【5】) return i; }
```

```
return 0;
```

```
}
```

1) A) s1[20], s2[20], \*p=s1, \*q=s2;

C) \*p=s1[20], \*q=s2[20];

2) A) gets(s1, s2);

C) gets(p, q);

3) A) index(\*p, \*q);

C) index(p, q);

4) A) qq=q;

C) ;

5) A) !\*qq

C) \*qq

B) \*p, \*q;

D) s1[ ], s2[ ], \*p=s1, \*q=s2;

B) gets(\*p, \*q);

D) gets(p), gets(q);

B) index(s1[ ], s2[ ]);

D) index(\*s1, \*s2);

B) pp=p;

D) pp=p, qq=q;

B) qq=='\0'

D) \*pp=='\0'

2、（10 分）以下程序名为 cat.c，其功能是将命令行中所列的文件依次在屏幕上显示，每显示完一个文件内容后，用户按任意键，则显示下一个文件，如果命令行中所列的文件不能打开或没有给出文件名，则显示相应的信息，命令的格式为：

cat filename1 filename2 ...

- 1) A) fgetc(fp) B) getchar(fp) C) fgetc(fp) D) getchar(fp)  
2) A) argv[] B) \*argv C) \*argv[] D) \*\*argv[]  
3) A) argc = 1 B) argc==1 C) argc==0 D) argv==1  
4) A) &fp B) \*fp C) argv[i] D) fp  
5) A) fclose(argv[i]) B) rewind(argv[i]) C) fclose(fp) D) rewind(fp)

```
#include"stdio.h"
void cat(FILE *fp)
{ int c;
  while((c= 【1  】)!=EOF)
    putchar(c);
}
main(argc,argv)
int argc;
char 【2  】;
{FILE *fp;
  int i;
  if( 【3  】 )
  { printf("缺少参数\n");exit(0);}
  for(i=1;i<argc;i++)
  { if((fp=fopen(argv[i],"r"))==NULL)
    printf(" %s 无法打开\n",argv[i]);
    else {
      printf("/*文件:%s */\n",argv[i]);
      cat( 【4  】 );
      【5  】 ;
      printf("按任意键显示下一文件\n");
      getchar();
    }
  }
}
```

#### 四 程序填空 (12 分)

1) 以下程序段用以统计链表中元素的个数。其中 first 指向链表第一个结点, count 用来统计结点个数, 请填空。

```
struct link
{ char data;
  struct link *next;
};
struct link *p, *first;
int count=0;
p=first;
while( 【 1 】 )
{ 【 2 】 ;
  p= 【 3 】 ;
}
```

2) 函数 fun() 的功能是将字符串 str 按照字符顺序从小到大进行排序, 并删除重复的字符, 然后返回该字符串。

【 4 】 fun( char \*str)

```
{ char *p, *q, *r, c;
  for(p=str; *p; p++) {
    r=p;
    for(q=p+1; *q; q++) if( 【 5 】 ) r=q;
    if(r!=p) { c=*r; *r=*p; *p=c; }
  }
  for(p=str; *p; p++) {
    for(q=p+1; *p==*q; q++);
    strcpy( 【 6 】 , q);
  }
  return str; }
```

#### 五、编程题(8 分)

请编写函数 void fun(int \*a, int y, int \*n), 它的功能是: 把 y 值插入到 a 数组中, a 数组中的数由主函数按由小到大的次序从键盘读入, y 是要插入的数, 其值也从键盘读入。n 是数组 a 中数字的个数。插入后, 数组中的数仍然有序。

```
#include <stdio.h>
#define M 50
void fun (int *a, int y, int *n)
{
  /*答案开始 */
  .....
  /*答案结束 */
}
```

一. 选择题

1-5: A 6-10:ACDCC 11-15: BCDEB 16-20:ABDCD

二.

1. 运行结果:

A k=2

B k=4

C k=7

E k=15

2.运行结果

4294967295

-1 3777777777

3.运行结果

39

0

4.运行结果

4354840

3

4354848

5

1

4

4354856

9