

## 《计算机软件技术基础2》(C++、64 学时)

(A 卷 共 4 页)

## 一. 单项选择题

- 以下说法中正确的是\_\_\_\_\_
 

A) C++程序总是从第一个定义的函数开始执行      B) C++程序总是从 main() 函数开始执行

C) main() 函数必须放在程序的开始部分      D) 一个函数中只允许使用一对花括号
- 设  $\text{int } i=2.8*6$ ;  $i$  定义后所赋的初值是\_\_\_\_\_
 

A) 16.8      B) 18      C) 17      D) 16
- 要求运算量必须是整数的运算符是\_\_\_\_\_
 

A) !      B) /      C) %      D) ++
- 表达式 'd' - 'a' + 2 的结果是
 

A) da2      B) 5      C) 2      D) 非法表达式
- `char s[]="ab23"`, `sizeof(s)` 和 `strlen(s)` 的结果分别是
 

A) 5 和 4      B) 5 和 5      C) 4 和 5      D) 4 和 4
- 以下指针用法中错误的是
 

A) `int a; int *p = &a;`      B) `int a=0, b=1; int *p = &a; *p=b;`

C) `int *p=3;`      D) `int a=0; *p = a; *p=3;`
- 设 `int a=1, b=15, c=-1`; 则表达式 `a>b>c` 的值为
 

A) 1      B) 0      C) 15      D) -1
- 下列关于 C++ 函数的叙述中, 正确的是
 

A) 每个函数至少要具有一个函数      B) 每个函数都必须有类型

C) 在一个函数的中可以定义另一个函数      D) 函数不能自己调用自己
- 关于对象, 下列说法中错误的是
 

A) 对象是类的一个实例      B) 任何一个对象只能属于一个类

C) 一个类只能有一个对象      D) 类与对象之前的关系和数据类型与变量之间的关系类似
- `int a[5]={-1, 1, 3, 5, 7}`, `*p=a+2`; 下列表达式中值为 3 的是
 

A) `*(a+2)`      B) `*(a+3)`      C) `*(p+2)`      D) `*p+1`
- 若定义: `int a[4]={1, 2, 3, 4, 5, 6, 7, 8, 9}`; 则数组 `a` 的第一维的大小是
 

A) 2      B) 3      C) 4      D) 不确定
- 下列是为字符数组赋字符串的语句组, 其中错误的是
 

A) `char s[10]; s="abc";`      B) `char s[]="abc";`

C) `char s[10]="abc"`      D) `char s[10]; strcpy(s, "abc");`
- 循环 `while(int i=3)i--`; 的执行次数是
 

A) 0      B) 1      C) 3      D) 无限
- 下列关于栈的叙述正确的是
 

A) 栈是非线性结构      B) 栈是一种树状结构

C) 栈具有先进先出的特征      D) 栈具有后进先出的特征
- 以下关于哈夫曼树的描述中不正确的是
 

A) 不存在度数为 1 的节点      B) 每个叶结点都有一个权值

C) 左子树中结点的值均小于根结点的值 D) 是最优二叉树

16. 设二叉树 B 中度数为 2 的结点个数为  $n_2$ , 则 B 中叶结点的个数为

A)  $n_2$  B)  $n_2 - 1$  C)  $n_2 + 1$  D) 不确定

17. 链表不具有的特点是\_\_\_\_\_

A) 可随机访问任一元素 B) 不必事先估计存储空间  
C) 插入删除不需要移动元素 D) 所需要空间与线性表长度成正比

18. 下列对派生类的描述中, 错误的是\_\_\_\_\_

A) 派生类中包含了它的基类的成员 B) 基类成员的访问权限在派生类中保持不变  
C) 一个派生类可以作另一个派生类的基类 D) 派生类至少有一个基类

19. 设  $\text{int } a=4$ ; ofstream  $ae("bb.txt")$ ; 则可以将 a 的值写入文件 bb.txt 中的是\_\_\_\_\_

A)  $\text{cout} << a$  B)  $aa << a$  C)  $bb << a$  D)  $aa.put(a)$

20. 实现运行时的多态性要使用\_\_\_\_\_

A) 构造函数 B) 析构函数 C) 重载函数 D) 虚函数

二、写出以下程序的运行结果

1. #include <iostream.h>

```
int fc(char*s, char c){
    if(s[0]==0) return 0;
    else if(s[0]==c) return 1+fc(s+1, c);
    else return fc(s+1, c);
}

void main(){char str[]="abba23b", cout<<fc(str, 'b')<<endl; }
```

程序运行的结果是:

2. #include <iostream.h>

```
#include <string.h>
void q(char *p){
    char *p1, *p2, c;
    for(p1=p, p2=p+strlen(p)-1; p1<p2; p1++, p2--){
        if(*p1>='0' && *p1<='9'){
            C=*p1; *p1=*p2; *p2=C;
            p1--; p2++;
        }
    }
}
```

```
void main(){
    char s[]="a234bcxy";
    q(s);
    cout<<s<<endl;
}
```

程序运行的结果是:

3. #include <iostream.h>

```
class goods{
    static int totalWeight;
```

```

    Int weight;
public:
    goods(int w){weight=w;totalWeight=w;}
    ~goods(){totalWeight-=weight;}
    Int getwg(){return weight;}
    static int getTotal(){return totalWeight;}
};

```

```

Int goods::totalWeight=0

```

```

void main(){
    cout<<goods::gettotal()<<endl;
    goods g1(38);
    goods g2(59);
    cout<<goods::getTotal()<<endl;
}

```

程序运行的结果是:

```

4.#include<iostream.h>
const int M=10;
void print(int*p,int n){
    for(int i=0;i<n;i++)cout<<p[i]<<" ";
    cout<<endl;
}
void main(){
    int a[M+1]={2,5,19,36,40,57,61,78,83,99};
    int b,i,p1=a,*p2;
    cin>>b;
    for(i=0;i<M+1;i++,p1++)
        if(*p1==b) print(a,M);break;
        else if(*p1>b){
            p2=a+M;
            for(;p2>p1;p2-)*p2=*(p2-1);
            *p1=b;
            print(a,M+1);
            break;
        }
}

```

若输入 70, 程序运行的结果是:

```

5.#include<iostream.h>
void fun(int &a,int b=3){
    static int i=2;
    a=a+b+i;
    i=i+a;
}
void main()
    int x=5,y=2;
    fun(x,y);
}

```

```
cout<<x<<" "<<y<<endl;
```

```
}
```

程序运行的结果是:

```
6.#include<iostream.h>
```

```
void main(){
```

```
int x=0,y=9;
```

```
if(++x>0)y++;
```

```
if(x!=0)y++;
```

```
elsey-;
```

```
cout<<x<<" "<<y<<endl;;
```

```
}
```

程序运行的结果是:

### 三、程序填空

1.以下程序实现将一个十进制整数转换成二进制数并输出。

```
#include<iostream.h>
```

```
class stack{
```

```
int s[10],top; //top 为栈顶指示器
```

```
public:
```

```
stack(){top=-1; //构造函数,把栈置为空栈
```

```
} bool Push(int x){ //入栈
```

```
if (top==9)return false; s[++top]=x; return true;
```

```
}
```

```
bool pop(int&x){ //出栈
```

```
if (_____) return false; //栈空,返回 false
```

```
x=_____; //出栈,并把栈顶元素存于 x
```

```
return true;
```

```
}
```

```
};
```

```
void main(){
```

```
stack stk;
```

```
int x,a;
```

```
cin>>x;
```

```
while(x!=0){
```

```
stk.Push(x%2);
```

```
x/=2;
```

```
}
```

```
while(_____)cout<<a;
```

```
cout<<endl;
```

2.把前 100 个素数写入文本文件“e:\primes.txt”.(提示:任何不是素数的整数都可以被更小的素数整除。)

```
#include<fstream.h>
```

```
#include<stdlib.h>
```

```
void main(){
```

```
long primes[100]={2,3,5}; //在 primes 数组中将存储前 100 个素数
```



```

int count=3;           //记录已找到的素数的个数
long trail=5;          //存储下一个要测试的整数
bool isprime=true;
do{
    trial+=2;
    int i=0;
    do{
        isprime= _____ ;
    }while(++i<count && isprime);
    if(isprime) primes[count++]=trial;
}while( _____ );
}
if(!outFile){ cout<<"Cannot open the file!<<endl;exit(1);}
for(int i=0;i<100;i++)
    outFile<<primes[i]<<" ";
outFile.close();
}

```

3.

以下程序实现整数的循环右移，即最后边的数移动到最左边，如 12345 循环右移后变为 51234.

```

#include<iostream.h>
int fun(int a){
    int x,b;
    b=a%10; a/=10;
    _____ ;
    while( _____ ){b*=10;a/=10;}
    x+=b;
    _____ ;
}
void main(){
    int a=12345;
    cout<<"循环右移后的结果为:" << _____ <<endl;
}

```

4. 以下函数寻找字符 ch 在字符串 s 中最后出现的位置；并在 s 中删除从该位置开始的所有字符。

```

#include<string.h>
void strch(char*s,char ch){
    for(int i=strlen(s)-1;i>=0; _____ )
        if( _____ ){
            s[i]=0;
            _____ ;
        }
}

```

5. 函数 (fun) 的功能是输出  $n(n < 10)$  阶杨辉三角。形式如下：

```

void fun(int a[][10],int n){
    int i,j;
}

```

```

1
1 1
1 2 1

```

```

for(i=0;i<n;i++)
a[i][0]=a[i][i]=_____ ;
for(i=2;i<n;i++)
    for(j=1;j<i;j++)
        a[i][j]= _____ ;
for(i=0;i<n;i++){
    for(j=0; _____ j++)
        cout<<a[i][j]<<" ";
    cout<<endl;
}
}

```

```

1 3 3 1
1 4 6 4 1
.....

```

#### 四、简答题

- 1.为简化处理，一般在单链表的开始结点之前增加一个头结点，使 head 恒指向头结点。请说明增加头结点的好处。
- 2.现有数字序列{45, 24, 53, 12, 37, 96, 30}，请按给出的顺序构造一颗二叉排序树，并给出其中序遍历序列。
- 3.一完全二叉树的顺序存储结构如下图所示，请分别指出 B 和 F 的双亲、左子女和右子女。

```

0 1 2 3 4 5 6 7 8 9 10 11
A B C D E F G H I J K L

```

- 4.请说明线性表、栈、队列的异同。

#### 五、编写程序

编写函数把一维整型数组中所有的非 0 元素放到数组的前半部分，并返回非 0 元素的个数。该函数的原型声明为：int fun(int a[],int n); 其中 n 为数组中包含的元素个数。

在主函数中从键盘向一个一维整型数组输入 10 个整数，调用 fun() 函数进行相应的处理，最后输出数组中所有的非 0 元素。

计算机软件技术基础2 答案

一、1~5 BDCBA 6~10 CABCA 11~15 BADDC 16~20 CABBD

二、1. 3

2. Ayxcb432

3. 0 97

4. 2 5 19 36 40 57 61 70 78 83 99

5. 9,2,23,2 6. 1,11

三、1. Top--1 s[top--] stk.pop(a)

2. Trial % primes[i] count<100 ofstream outFile"e:\\primes.txt"

3. X=a al=0 return x fun(a)

4. i-- s[i]=ch return

5. 1 a[i-1][j-1]+a[i-1][j] j<=i

四、1.增加头结点后，使在表中任意位置插入删除操作都无区别，无须做特殊处理，对空表和非空表的处理也相同

2、得一由小到大的排序 45

24 53  
12 37 96  
30

3.B: DE F.L 无右子女

4.栈和队列都是运算受限的线性表,栈是规定只能在一端进行插入和删除运算的线性表,队列是规定只能在一端进行插入而在另一端进行删除的线性表。

五、#include <iostream.h>

```
int fun(int a[],int n){
    int i=0,j;
    while (i<n && a[i]!=0)i++;
    j=i+1;
    while(i<n){if (a[i]!=0)
        {a[i]=a[j];a[j]=0;i++;j++;}
    }
    return i;
}
```

```
void main( ) {int x[10],n;
    for (int i=0;i<10;i++) cin>>x[i];
    n=fun(x,10);
    for(i=0;i<n;i++)
        cout<<x[i]<<" ";
    cout<<endl;
}
```