

# 2014~2015 学年第一学期期末考试试卷

## 《计算机软件技术基础 2》(C++、64 学时)(A 卷 共 4 页)

(考试时间: 2015 年 1 月 22 日)

题号	一	二	三	四	五	成绩	核分人签字
得分							

### 一. 单项选择题(每题 1 分, 共 20 分)

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
答案																				

- 能将 C++ 语言源程序转换为目标程序的是 \_\_\_\_。  
A) 连接程序      B) 解释程序      C) 编译程序      D) 编辑程序
- 设 `int i=2.8*6;` 则变量 `i` 的值是 \_\_\_\_。  
A) 12      B) 16      C) 17      D) 18
- `a`、`b` 均为 `int` 型且 `b≠0`, 则表达式 `a-a/b*b` 的值是 \_\_\_\_。  
A) 0      B) `a`      C) `a` 被 `b` 除的余数部分      D) `a` 被 `b` 除商的整数部分
- 字符串 `"\\nd:\\fl.txt"` 的存储长度是 \_\_\_\_。  
A) 10      B) 11      C) 12      D) 13
- 程序中语句执行的顺序有三种基本控制方式, 它们分别是分支、循环和 \_\_\_\_。  
A) 跳转      B) 顺序      C) 递归      D) 返回
- 设 `int i,k; for(i=0,k=-1;k=i; i++,k++);` 对以上 `for` 循环语句, 正确的说法是 \_\_\_\_。  
A) 编译报错      B) 循环一次      C) 无限循环执行空语句      D) 一次也不循环
- 若有语句 `int k,*p=&k;` 则以下与这个语句等效的语句序列是 \_\_\_\_。  
A) `int k,*p; p=&k;`      B) `int k,*p; p=k;`      C) `int k,*p; *p=&k;`      D) `int k,*p; *p=k;`
- 设 `int a[5][5], *p1=a[0], (*p2)[5]=a+1;` 则以下不能用于访问数组元素 `a[3][2]` 的是 \_\_\_\_。  
A) `*(a[3]+2);`      B) `p1[17]`      C) `p2[3][2]`      D) `p2[2][2]`
- 设 `char s[10]="program", t[]="test";` 则下列选项中错误的是 \_\_\_\_。  
A) `strcpy(s,t);`      B) `s=t;`      C) `*s=t[1];`      D) `*(t+2)='w';`
- 下列关于函数的叙述中, 正确的是 \_\_\_\_。  
A) 每个函数必须有返回类型      B) 每个函数至少要有个形参  
C) 函数不能自己调用自己      D) 对一个函数的调用必须在它的定义之后

11. 设 `int *a=new int [15];` 要释放该动态申请的一维数组空间应使用的语句是 \_\_\_\_\_。
- A) `delete int[15];`      B) `delete []a;`      C) `delete a[15];`      D) `delete *a;`
12. 下列关于局部静态变量的叙述中错误的是 \_\_\_\_\_。
- A) 具有块作用域      B) 其缺省的初始值为 0      C) 具有静态生存期      D) 具有文件作用域
13. 设 AB 为一个类, 执行 “`AB a(4),b(a),*p=&a;`” 时, 自动调用 AB 类的构造函数和拷贝构造函数的次数分别是 \_\_\_\_\_。
- A) 2 次和 1 次      B) 1 次和 1 次      C) 3 次和 0 次      D) 3 次和 1 次
14. 生成派生类对象时, 派生类构造函数调用基类构造函数的条件是 \_\_\_\_\_。
- A) 无需任何条件      B) 基类中显式定义了构造函数  
C) 派生类中显式定义了构造函数      D) 派生类构造函数明确调用了基类构造函数
15. 在类中声明虚函数的目的是实现 \_\_\_\_\_。
- A) 封装性      B) 继承性      C) 静态多态性      D) 动态多态性
16. 如果想向文本文件 `fl.txt` 中输出信息, 则定义相关流类对象 `myfile` 的语句是 \_\_\_\_\_。
- A) `ifstream myfile("fl.txt");`      B) `ofstream myfile("fl.txt");`  
C) `infile myfile("fl.txt");`      D) `outfile myfile("fl.txt");`
17. 数据的逻辑结构与数据元素本身的内容和形式 \_\_\_\_\_。
- A) 可能相关      B) 可能无关      C) 一定相关      D) 一定无关
18. 哈夫曼树是 \_\_\_\_\_。
- A) 最优二叉树      B) 完全二叉树      C) 满二叉树      D) 二叉排序树
19. 设某二叉树的先序和中序遍历序列分别为 `CABD` 和 `ABCD`, 则其后序遍历序列是 \_\_\_\_\_。
- A) `CABD`      B) `BCDA`      C) `CBDA`      D) `BADC`
20. 二分查找时, 查找表中所有数据元素必须按关键字值 \_\_\_\_\_ 排列。
- A) 递增顺序      B) 递减顺序      C) 有序      D) 任意顺序

## 二. 写出以下程序的运行结果 (每题 4 分, 共 24 分)

```
1. #include <iostream.h>
   void main(){
       int x=10,y=9,a,b,c;
       a=(--x==y++)?--x:++y;
       b=x++;
       c=y;
       cout<<"a="<<a<<"  b="<<b<<"  c="<<c<<endl;
   }
```

程序运行的结果是:



```

2.  #include <iostream.h>
    int sum( int p[ ], int & s ){
        int i, sc=0;
        for(i=0; i<3; i++){ sc+=p[i]; s+=p[i]; }
        return sc;
    }
    void main( ){
        int data[][3]={1,2,3,4,5,6,7,8,9 },s = 0;
        for(int i=0; i<3; i++) { cout<<sum(data[i],s)<<" "; cout<<s<<endl; }
    }

```

程序运行的结果是：

```

3.  #include <iostream.h>
    #include <string.h>
    void fun(char *a,int n){
        char ch;
        for(int i=0;i<n/2;i++){ch=a[i]; a[i]=a[n-i-1]; a[n-i-1]=ch;}
    }
    void main(){
        char s1[]="abcdef",s2[]="123456789";
        fun(s1,strlen(s1)); fun(s2+3,4);
        cout<<s1<<endl<<s2<<endl;
    }

```

程序运行的结果是：

```

4.  #include <iostream.h>
    void main() {
        int i=5;
        do{
            switch(i%2){
                case 0:i--;break;
                case 1:i--;continue;
            }
            i--; cout<<i;
        }while (i>0);
        cout<<endl;
    }

```

程序运行的结果是：

```

5.  #include <iostream.h>
    class Point{
        int X,Y;
    public:
        Point(int xx=0,int yy=0){X=xx; Y=yy; cout<<"Point constructor\n";}
        ~Point(){cout<<"Point distructor\n";}
        void move(int xoff,int yoff){X+=xoff; Y+=yoff;}
        int GetX(){return X;}
        int GetY(){return Y;}
    };
    class Rectangle:public Point{
        int W,H;
    public:
        Rectangle(int x,int y,int w,int h):Point(x,y),W(w),H(h){ cout<<"Rectangle constructor\n";}
        ~Rectangle(){cout<<"Rectangle distructor\n";}
        int GetW(){return W;}
        int GetH(){return H;}
    };
    void main(){
        Rectangle r1(3,5,8,20);
        r1.move(2,6);
        cout<<"The Data of r1(X,Y,W,H):\n";
        cout<<r1.GetX()<<","<<r1.GetY()<<","<<r1.GetW()<<","<<r1.GetH()<<endl;
    }

```

程序运行的结果是：

```

6.  #include <iostream.h>
    void fun(int x,int y,int *z){ *z=x+y;  x++;  y++; }
    void main() {
        int a(2),b(5),c(8);
        fun(a,b,&c);  fun(c,a,&b);  fun(b,c,&a);
        cout<<a<<" "<<b<<" "<<c<<endl;
    }

```

程序运行的结果是：

### 三. 程序填空 (每空 2 分, 共 32 分):

1. 以下程序中, 函数 `uppers()` 的功能是, 把由 `s` 指向的字符串中所有小写英文字母转换成大写并保存到由 `us` 指向的存储空间中。程序的输出结果是 "THANK YOU!". 请填空完成该程序。

```
#include<iostream.h>
void uppers(char *s,char *us){
    for(;*s!='\0';s++, _____ )
        if( _____ ) *us=*s-32;
        else *us=*s;
        _____ ;
}
void main() {
    char s1[20]="Thank You!",s2[20];
    _____ ;
    cout<<s2<<endl;
}
```

2. 以下程序的功能是, 重排数组 `a` 中的元素, 使相等元素放在一起, 并保持它们在数组中第一次出现的相对次序, 程序输出结果为: 8 8 8 8 5 5 5 2 2 3 3 1 程序采用的算法是: 先删除 `a` 中重复元素, 并统计各不同元素出现次数保存到数组 `c`, 再按各元素重复次数写回到数组 `a` 中。

```
#include<iostream.h>
_____ N=12;
void main() {
    int a[N]={8,5,8,2,8,3,5,2,8,3,5,1},b[N],c[N],i,j,k,t;
    for(i=0;i<N;i++) b[i]=c[i]=1;
    k=-1;
    for(i=0;i<N;i++)
        if( _____ ){
            a[++k]=a[i];
            for(j=i+1;j<N;j++)
                if(a[j]==a[k]){ b[j]=0; _____ ;}
        }
    t=N-1;
    for(i=k;i>=0;i--)
        for(j=1;j<=c[i];j++) a[t--]= _____ ;
    for(i=0;i<N;i++) cout<<a[i]<<" ";
    cout<<endl;
}
```

3. 以下程序的功能是, 输出 1 至 100 之间每位数的乘积大于每位数的和的数。

```

#include<iostream.h>
void main() {
    int n,k,s,m;
    for(n=1;n<=100;n++){
        k=1; s=0;
        _____ ;
        while( _____ ){
            k*=m%10; s+=m%10;
            _____ ;
        }
        if(k>s) cout<<n<<" ";
    }
    cout<<endl;
}

```

4. 以下程序的输出结果为 7/12,。请填空完成该程序。

```

#include<iostream.h>
int GCF(int x,int y){ int r; while(r=x%y){ x=y;y=r; } return y; } //求"x,y 的最大公约数
class Fra{ //分数类
    int num,den; //分别表示分数的分子和分母
public:
    Fra(int n=0,int d=1){ num=n; den=(d!=0)?d:1; }
    void show(){ cout<<num<<"/"<<den<<endl; }
    void simp(){ int r= _____ : num/=r; den/=r; } //化简分数
    _____ (Fra &fra){ //重载加法的成员函数
        Fra sum;
        sum.den=den*fra.den; sum.num=num*fra.den+den*fra.num;
        sum.simp(); return sum;
    }
};
void main() { Fra fra1(-1,4),fra2(5,6),fra3; fra3=fra1+fra2; fra3.show(); }

```

5. 带头结点的字符型单链表类及其相应结点类的声明如下。成员函数 Del()的功能是，删除所有结点值为 ch 的结点。

```

class Node{
    friend class Chain;
    int data;
    Node *next;
};

```



```

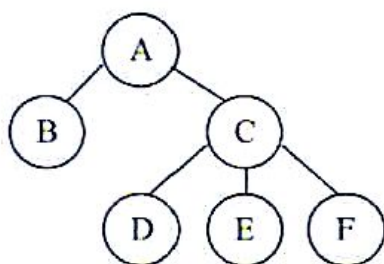
class Chain{
    Node *head;           //头指针
public:
    Chain(){ head=new Node; head->next=0; }    //构造函数
    void Del(char ch){
        Node *p= _____, *q;
        while(p->next!=0){
            q=p->next;
            if(q->data==ch){
                p->next=q->next;
                _____;
            }
            else _____;
        }
        .....
    };
};

```

#### 四. 简答题 (12 分)

1. 设有 5 个元素 A、B、C、D、E 依次入栈，如果元素 C、D 最先出栈（即 C 第一个出栈，D 第二个出栈），则可能的出栈次序有哪几个？（3 分）

2. 请指出如下所示树的深度和度，并将其转换为二叉树。（设根结点的层数是 0）。（3 分）



3. 对整数序列 {45,80,48,40,22,78} 按升序排序，请分别给出在直接选择排序和直接插入排时对整数序列第 3 次处理后的结果。（3 分）

第 3 次直接选择排序后的结果是：

第 3 次直接插入排序后的结果是：

4. 设待散列序列为 {23,26,34,56,15,20,29}，散列函数为  $h(k)=k\%11$ ，请指出那些元素会发生冲突？为什么？（3 分）

#### 五. 编写程序。（12 分）

编写函数 `double fun(int n)`，其功能是：求  $n$  以内（不包括  $n$ ）同时能被 5 与 11 整除的所有自然数之和的平方根，并作为函数值返回。在 `main()` 函数中，输入一个自然数（如果不是自然数，需重新输入，直到满足要求为止），调用 `fun()` 求出相关结果并输出。



计算机计算机软件技术基础 2 (C++、64 学时) 试卷 A 参考答案 2015.1

一、

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

二、

1. a=8 b=8 c=10

2. 6 6  
15 21  
24 45

3. fedcba  
123765489

4. 20

5. Point constructor  
Rectangle constructor  
The Data of r1(X,Y,W,H):  
5, 11, 8, 20  
Rectangle distructor  
Point distructor

6. 16 9 7

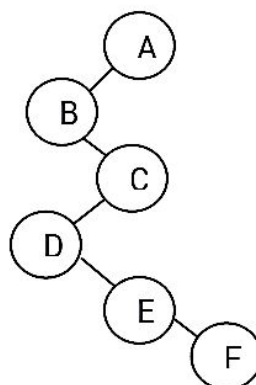
三、(有些答案不唯一)。

- us++ \*s>='a'&&\*s<='z' \*us=0 uppers(s1,s2)
- const int b[i]==1 c[k]++ a[i]
- m=n m!=0 m/=10
- GCF(num,den) Fra operator+
- head delete q p=p->next

四、

1. CDEBA CDBAE CDBEA

2. 树的深度: 2 树的度数: 3



3. 第 3 次直接选择排序后的结果是: {22,40,45,80,48,78}  
第 3 次直接插入排序后的结果是: {40,45,48,80,22,78}
4. 23、34、56 会发生冲突, 因为他们的散列的地址均为 1  
26、15 会发生冲突, 因为他们的散列的地址均为 4

五、

```
#include <iostream.h>                //程序结构 2 分
#include <math.h>                      //1 分
double fun(int n){                   //6 分
    double s(0);
    for(int i=1;i<n;i++)
        if(i%5==0&& i%11==0)
            s+=i;
    s=sqrt(s);
    return s;
}
void main(){                          //3 分
    int n;
    do{
        cout<<"请输入一个自然数: ";
        cin>>n;
    }while(n<1);
    cout<<fun(n)<<endl;
}
```