

2013~2014 学年第二学期期末考试试卷

《计算机软件技术基础 2》(C++、64 学时)(A 卷 共 4 页)

(考试时间: 2014 年 6 月 30 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																				

- 以下可以作为 C++ 标识符使用的是 D。
A) 3D B) byte-size C) double D) CHINA
- 已知 double x=3.6, y=5.2; 则表达式 (int)x+y 的值是 C。
A) 8.8 B) 8.6 C) 8.2 D) 8
- 设 char ch; 以下语句中错误的是 A。
A) ch="A"; B) ch='\w'; C) ch=65; D) ch='A'+32;
- 以下对 C++ 语言中关键字 default 的说法中正确的是 B。
A) 只能在循环体内使用 B) 最多在 switch 语句中使用一次
C) 可在 switch 语句中多次使用 D) 对其使用没有限制
- 设 int i, k; for(i=3, k=0; i<10; k+=i, i+=3); cout<<k; 该程序段的输出结果是 B。
A) 0 B) 3 C) 18 D) 27
- 设 char a[]="abcde"; 则 sizeof(a) 的值是 C。
A) 0 B) 5 C) 6 D) 不确定
- 设 int *ip; 为动态申请一个含 10 个元素的整型一维数组并将该数组首元素的地址存于 ip, 应使用的语句是 D。
A) *ip=new int[10]; B) ip=int[10]; C) int ip[10]; D) ip=new int[10];
- 已知 int x=10, *p=&x; 则 p 的值是 B。
A) 10; B) x 的地址 C) x 的别名 D) 不确定
- 已知 int a[10], *p; p=a; 则以下表达式中错误的是 D。
A) p+=2 B) a+=2 C) p-a D) a+2
- 通过函数原型不能了解到的是 D。
A) 函数的类型 B) 函数参数的个数 C) 函数参数的类型 D) 函数的功能

11. 在函数形参表中定义的变量是 B。

- A) 全局变量 B) 局部变量 C) 静态变量 D) 寄存器变量

12. 设已定义函数 `int max(int x, int y)` 其功能是返回两个数中较大的一个, 若要调用该函数求 15 28 36 三者中最大值, 以下不正确的是 D。

- A) `int m=max(max(15,28),36);` B) `int m=max(36, max(15,28));`
C) `int m=max(max(36,28),15);` D) `int m=max(15,28,36);`

13. 设函数模板的定义为: `template <class T> T fun(T x,T y){return x*x+y*y;}` 在下列对 fun 的调用中错误的是 B。

- A) `fun(3,8)` B) `fun(5,12.3)` C) `fun(7.0,4.6)` D) `fun<double>(4,2.8)`

14. 以下关于析构函数的叙述中正确的是 C。

- A) 其名与类名完全相同 B) 其类型是 void C) 不可重载 D) 函数体中必须有 delete

15. 如果在一个公有派生类的成员函数中不能直接访问从基类继承下来的某个成员, 则该成员一定是基类中的 B。

- A) public 成员 B) private 成员 C) protected 成员 D) private 或 protected 成员

16. 虚函数在实现 A 时使用。

- A) 动态多态性 B) 静态多态性 C) 函数重载 D) 运算符重载

17. 在输入/输出时使用的 `cin` 和 `cout` 是 A。

- A) 对象名 B) 类名 C) 函数名 D) 常量名

18. 设指针 `p` 指向单链表中的结点 `m`, 要删除 `m` 之后的一个结点 (假设其存在), 需要执行的操作是 C。

- A) `p->next=p;` B) `p=p->next;` C) `p->next=p->next->next;` D) `p=p->next->next;`

19. 在一棵二叉树上第 4 层 (设根结点的层数为 0) 的结点数最多是 C。

- A) 4 B) 8 C) 16 D) 32

20. 对关键字序列 {19, 14, 23, 1, 68, 20, 84, 27, 55, 11, 10, 79} 进行哈希存储时, 使用的哈希函数为 $h(k)=k\%13$, 并用链地址法解决冲突, 则哈希表中与地址 1 对应的单链表中结点的个数是 D。

- A) 1 B) 2 C) 3 D) 4

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(){`

`char c1('3'),c2('5');`

`// 字符 '0' 的 ASCII 值为 48`

`int n1,n2;`

`n1=c1; n2=c2-c1;`

`cout<<n1<<" "<<n2<<endl;`

`n1=c1-'0'; n2=10*n1+(c2-'0');`

`cout<<n1<<" "<<n2<<endl;`

`}`

程序运行的结果是:


```

2.  #include <iostream.h>
    void main(){
        int a,q,n,sum;
        a=1;q=2;n=0,sum=0;
        do{ sum+=a; ++n;  a*=q; }while(sum<50);
        n-=1;
        cout<<"sum="<<sum<<"  n="<<n<<endl;
    }

```

程序运行的结果是：

```

3.  #include <iostream.h>
    class Pnt{
        double x,y;
    public:
        Pnt(double a=0,double b=0){ x=a;y=b;}
        void show(){cout<<"("<<x<<","<<y<<")\n";}
    };
    class Crc:public Pnt{
    public:
        Crc(){Rds=1;}
        Crc(double a,double b,double r=1):Pnt(a,b),Rds(r){ }
        void show(){ cout<<Rds;  Pnt::show(); }
    private:
        double Rds;
    };
    void main(){
        Crc  c1,c2(5,5),c3(2.6,3.7,5.8);
        cout<<"圆 c1:  "; c1.show();
        cout<<"圆 c2:  "; c2.show();
        cout<<"圆 c3:  "; c3.show();
    }

```

程序运行的结果是：

```

4.  #include <iostream.h>
    void main(){
        char str[]="I need your help.";
        int count(0),flag(0),i;
        for(i=0;str[i]!='\0';i++)
            if(str[i]==' ')flag=0;      //' ' 为空格符
            else if(flag==0){ flag=1;  count++; }
        cout<<i<<" " <<count<<endl;
    }

```

程序运行的结果是：

```

5.  #include <iostream.h>
    void fun(int *a,int n,int *im){
        *im=0;
        for(int i=1;i<n;i++)  if(a[i]<a[*im]) *im=i;
    }
    void main(){
        int x[3][4]={6,12,5,44,18,9,21,15,34,22,37,61};
        for(int i=0;i<3;i++){ fun(&x[i][0],4,&j); cout<<i<<" " <<j<<" " <<x[i][j]<<endl; }
    }

```

程序运行的结果是：

```

6.  # include <iostream.h>
    void p(int n,int b){
        if(n){
            p(n/b,b);
            cout<<n%b;
        }
    }
    void main(){
        p(10,2); cout<<endl;
        p(80,8); cout<<endl;
    }

```

程序运行的结果是：

三. 程序填空 (每空 2 分, 共 32 分):

1. 以下程序的运行结果为: 1 3 5 7 9 11 13 15, 请填空完成该程序。

```
#include <iostream.h>
const int N=80;
class stack{                //整型顺序栈类
    int s[N],top;
public:
    stack(){top=-1;}
    bool Push(const int &x){    //进栈
        if(top==N-1) return false;
        s[ ]=x;    return true;
    }
    bool Pop(){                //出栈
        if(top== -1) return false;
        ;    return true;
    }
    bool Top(int &x){          //将栈顶元素取至 x
        if(top== -1) return false;
        x= ;    return true;
    }
};
void main(){
    stack s1; int i,a;
    for(i=15;i>0;i-=2) if(!s1.Push(i)) break;
    while( ){    cout<<a<<" "; s1.Pop(); }
    cout<<endl;
}
```

2. “水仙花数”是指一个三位数, 其各位数的立方和等于该数。例如: $153=1^3+5^3+3^3$ 。以下程序的功能是, 对输入的一个三位正整数判断其是否为“水仙花数”, 并输出相应结果。

```
#include <iostream.h>
int fun(int n){
    int i,j,m,k;
    m=n; ;
    for(i=1;i<4;i++){ ; m=(m-j)/10; k=k+j*j*j; }
    if(k==n) return 1; else return 0;
}
```

```

void main(){
    int a;
    do{ cout<<"请输入一个三位正整数: "; cin>>a; }while( _____ );
    if( _____ ) cout<<a<<"是水仙花数\n";
    else cout<<a<<"不是水仙花数\n";
}

```

3. 设已存在文本文件 a.txt, 以下程序的功能是, 最多从 a.txt 中提取 20 个字符, 将它们按 ASCII 值由小到大的顺序输出到文本文件 b.txt。

```

#include <iostream.h>
#include <fstream.h>
void main(){
    _____ s1("e:\\a.txt");
    ofstream s2("e:\\b.txt");
    if(!s1||!s2){ cout<<"不能打开文件\n"; return;}
    char ch,t[20];    int i,count(0);
    while(s1.get(ch)){
        for(i=count-1;i>=0&&ch<t[i];i--) t[i+1]=t[i];
        t[i+1]=ch;    count++;
        if(count==20) _____ ;
    }
    for(i=0;i<count;i++) _____ <<t[i];
    s1.close(); s2.close();
}

```

4. 以下程序的运行结果为: 7 9 11 13 15, 请填空完成该程序。

```

#include <iostream.h>
const int N=5;
class array{
    int a[N];
public:
    array( _____ ){ for(int i=0;i<N;i++) a[i]=i+n;}
    _____ ;
    void print(){ for(int i=0;i<N;i++) cout<<a[i]<<" "; cout<<endl; }
};
void operator+=(array &a1,array &a2){ for(int i=0;i<N;i++) a1.a[i]= _____ ;}
void main(){ array v1(2), v2(5); v1+=v2; v1.print();}

```


5. 以下函数的功能是，把由 s2 指向的字符串连到由 s1 指向的字符串的后面。

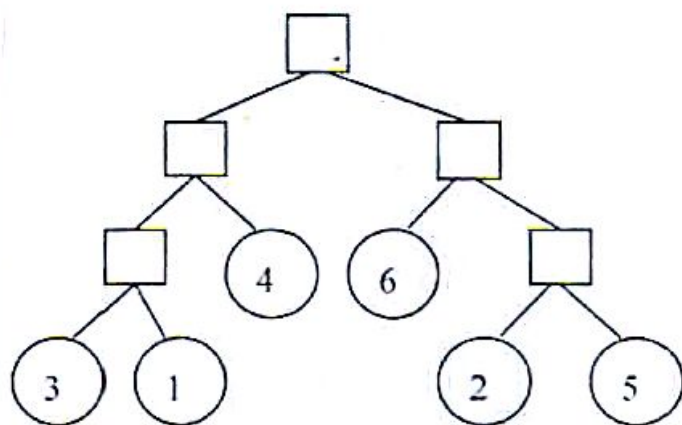
```
void strcat(char *s1, char *s2){  
    while(*s1) s1++;  
    while(*s2){ *s1=*s2; s1++; _____ ;}  
    _____ =0;  
}
```

四. 简答题 (12 分)

1. 要使存储线性表的空间大小与线性表的长度保持一致，应使用何种存储方式，为什么？ (3 分)

2. 已知关键码集合 {60, 70, 20, 40, 30, 90, 80, 10}，请按给出顺序构造二叉排序树。对该二叉排序树进行何种遍历，方可得到关键码的有序序列？ (3 分)

3. 画出与下面扩充二叉树对应的一颗最优扩充二叉树并计算其 WPL 值。 (3 分)



4. 用冒泡排序法对整数序列{19, 14, 23, 5, 68, 20}按升序排序, 最少需要进行几趟冒泡排序? 对 n 个待排序数据用冒泡排序法进行升序排序时, 在何种情况下使用的时间最少? 在何种情况下使用的时间最多? (3 分)

五. 编写程序。(12 分)

编写函数 `void del_dchar(char *p, int i)`, 其功能是删除字符串 `p` 中指定位置 `i` 的字符。编写主函数实现如下功能: 从键盘输入长度不超过 79 的字符串(可含空格), 通过调用 `del_dchar()` 函数删除该字符串中的所有数字字符, 并在屏幕上输出删除后的字符串。如果输入的字符串中无数字字符, 则输出“输入的字符串中无数字字符!”。

一、

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

二、

1. 51 2
 3 35



2. sum=63 n=5

3. 圆 c1: 1(0,0)
 圆 c2: 1(5,5)
 圆 c3: 5.8(2.6,3.7)

4. 17 4

5. 0 2 5
 1 1 9
 2 1 22

6. 1010
 120

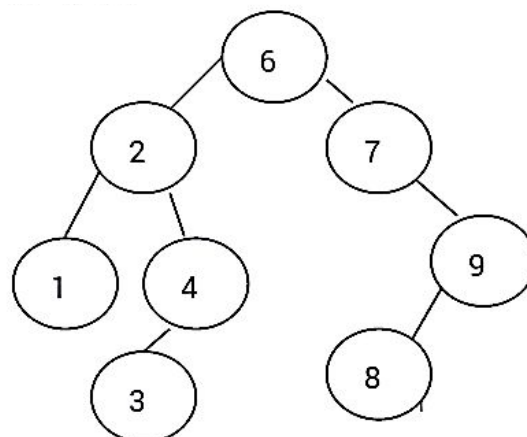
三、(有些答案不唯一)。

1. ++top top-- s[top] s1.Top(a)
 2. k=0 j=m%10 a<100||a>999 fun(a)
 3. ifstream break s2
 4. int n friend void operator+=(array &,array&) a1.a[i]+a2.a[i]
 5. s2++ *s1

四、

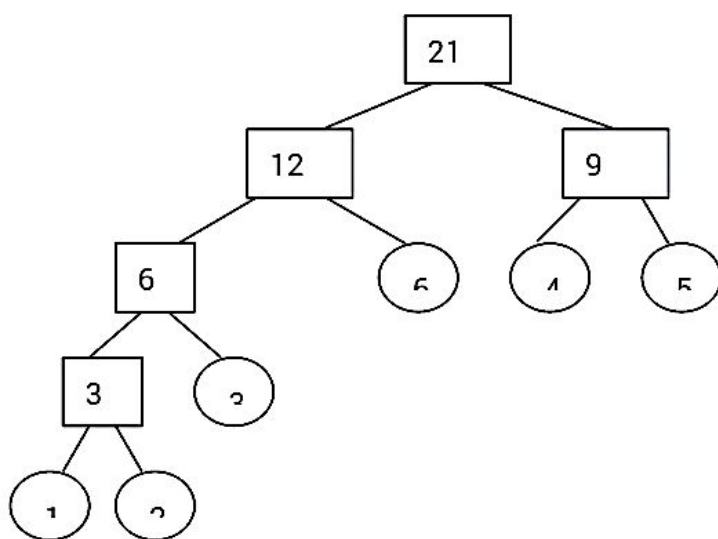
1. 链接存储方式。(1分) 存储线性表元素的结点空间可以动态申请和释放。(2分)

2. 二叉排序树:(2分)



中序遍历。(1分)

3. 最优扩充二叉树为：(2分)



$$WPL=1*4+2*4+3*3+4*2+5*2+6*2=51 \quad (1 \text{ 分})$$

4. 3趟

排序前待排数据已按排序码升序排列(正序排列)为最好情形。(1分)

排序前待排数据已按排序码降序排列(反序排列)为最坏情形。(1分)

五、 #include <iostream.h>

void del_dchar(char *p,int i);

void main(){

char str[80];

cout<<"输入长度不超过 79 的字符串(可含空格):";

cin.getline(str,80);

int i=0,flag=0;

while(i<80&&str[i]){

if(str[i]>='0'&&str[i]<='9')

{del_dchar(str,i);flag=1;}

else

i++;

}

if(flag)

cout<<"删除数字字符后的字符串: "<<str<<endl;

else

cout<<"输入的字符串中无数字字符! \n";

}

void del_dchar(char *p,int i){

for(int j=i;j<80&&p[j];j++) p[j]=p[j+1];

}

//程框架 1 分

//1 分

//1 分

//1 分

//3 分

//2 分

//3 分