

## 华北计算技术研究所 2005 年专业课试题

- 要求：
1. 答案必须写在答题纸上，并标明题号。
  2. 答案要字迹清楚，语义确切。
  3. 所有计算要求给出计算过程。

### 一、 填空题（20 分）

1. \_\_\_\_\_是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每一条指令表示一个或多个操作。它具有 5 个重要特征：\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。
2. 一棵非空的二叉树，其第  $i$  层上最多有\_\_\_\_\_个结点。满二叉树是一棵深度为  $k$  且恰好有\_\_\_\_\_个结点的二叉树。
3. 图的存储结构包括\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_等几种。图的遍历路径包括\_\_\_\_\_和\_\_\_\_\_。
4. 常用的构造哈希函数的方法有\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。

### 二、 选择题（20 分）

请在你认为正确的答案所对应的字母上画“√”。

1. 在 C 语言中，要存储一个 8 个字符的字符串，至少需要声明大小为多少的一维字符数组？  
(A) 7      (B) 8      (C) 9      (D) 10
2. 两个矩阵  $A: m \times n$ ,  $B: n \times p$  相乘，其时间复杂度为：  
(A)  $O(n)$       (B)  $O(mnp)$       (C)  $O(n^2)$       (D)  $O(n^3)$
3. 下列程序为将一条数据插入栈上：  

```
void add(int top,element item)
{ if (top>=MAX_STACK_SIZE-1)
  return stack_full();
```

stack[\_\_\_\_]=item;}

则在 stack[\_\_\_\_]的中括号内横线上的正确内容应为:

- (A) ++\*top      (B) \*top++      (C) \*top--      (D) \*top

4. 有如下函数:

void fun(struct node h1,struct node h2)

```
{ struct node *t;
  t=h1;
  while(t->next!='\0')
    t=t->next;
  t->next=h2;
}
```

其中形参 h1 和 h2 分别指向 2 个不同链表的第一个结点, 此函数的功能是:

- (A) 将链表 h2 接到链表 h1 后  
(B) 将链表 h1 接到链表 h2 后  
(C) 找到链表 h1 的最后一个结点由指针返回  
(D) 将链表 h1 拆分成两个链表

5. 一个栈的入栈序列是 abcde, 则栈的不可能输出序列是:

- (A) edcba                      (B) decba  
(C) dceab                      (D) abcde

### 三、 回答问题, 并给出理由。(10 分)

1. 设在一个有关串的程序编码当中, 有如下定义与赋值:

```
const char A[]={ 'a','b','c','\0' };
char B[]={ 'a','b','c','d','\0' };
... ..
```

```
for(i=0;i<4;i++)
```

```
{
```

```
  A[i]='a';
```

```
  B[i]='b';
```

```
}
```

```
... ..
```

```
...
```

```
...
```

在该程序编码中是否有错? 为什么?

2. 若 A 为一下三角矩阵数组, 则采用以行为主和以列为主的数据存放方式哪一种更合适? 为什么?

### 四、 根据要求编写算法。(20 分)

1. 线性表 A 和 B 均是按元素值递增有序排列, 均以单链表作存储结构。请编写

一算法将表 A 和表 B 归并成一个按元素值递减有序排列的线性表 C (允许表中含有值相同的元素), 并要求利用原表空间。

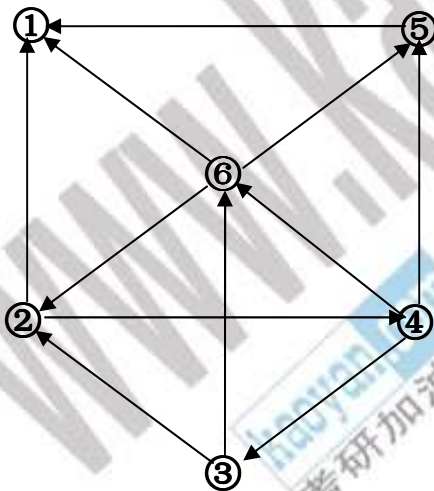
2. 编写一个算法, 对于输入的十进制非负整数, 将它的八进制表示打印出来。

### 五、 回答以下问题, 并给出计算或推理过程。(20 分)

1. 已知一棵二叉树的中序序列为 cbedahgijf, 后序序列为 cedbhjigfa。给出其先序序列, 并画出该二叉树。
2. 画出对长度为 10 的有序表进行折半查找的一棵判定树, 并求其等概率时查找成功的平均查找长度。

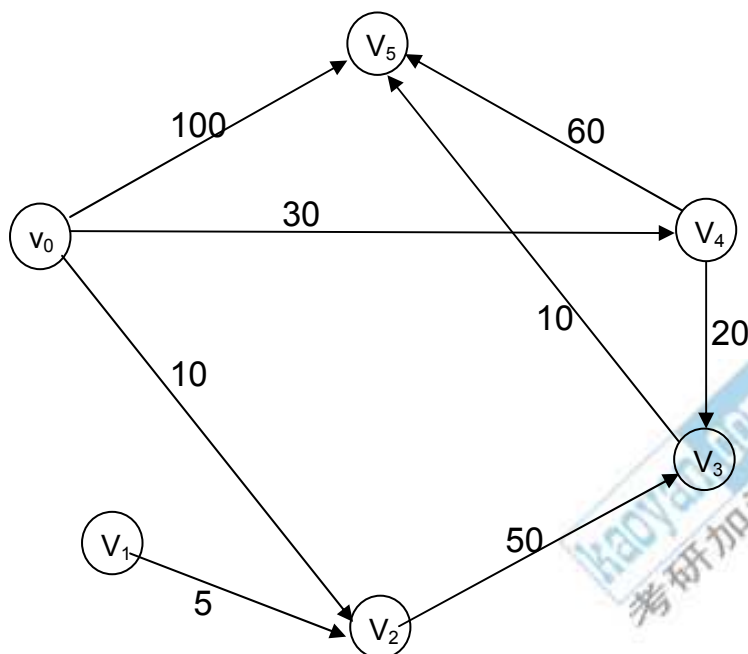
### 六、 已知如图所示的有向图, 请给出该图的:

- (1) 每个顶点的入/出度;
- (2) 邻接矩阵;
- (3) 邻接表;
- (4) 逆邻接表;
- (5) 强连通分量。(10 分)



### 七、 下图是一个有向图, 其中每条弧段上的数字表示该弧段的权值。

1. 请用 Dijkstra 算法计算  $v_0$  到各点的最短路径 (要求给出计算过程)。
2. 给出用 C 语言描述的 Dijkstra 算法。(20 分)



#### 八、 请回答以下有关 C++ 语言的问题。(16 分)

1. 请比较一下值调用与引用调用的相同点和不同点。
2. 什么叫作抽象类？抽象类有何作用？抽象类的派生类是否一定要给出纯虚函数的实现？
3. 什么叫作指针？指针中存储的地址和这个地址中的值有何区别？
4. 什么叫拷贝构造函数？拷贝构造函数何时被调用？

九、 建立基类 **Building**，用来存储一座楼房的层数、房间数以及它的总平方英尺数。建立派生类 **Housing**，继承 **Building**，并存储下面的内容：卧室和浴室的数量。另外，建立派生类 **OfficeBuilding**，继承 **Building**，并存储灭火器和电话的数目。  
然后，用 C++ 语言编制应用程序，建立住宅楼对象和办公楼对象，并输出它们的有关数据。(14 分)

## 华北计算技术研究所 2005 年专业课试题参考答案

### 十、 填空题 (20 分)

5. 算法 是对特定问题求解步骤的一种描述, 它是指令的有限序列, 其中每一条指令表示一个或多个操作。它具有 5 个重要特征: 有穷性、确定性、可行性、输入、输出。
6. 一棵非空的二叉树, 其第  $i$  层上最多有  $2^{i-1}$  个结点。满二叉树是一棵深度为  $k$  且恰好有  $2^k-1$  个结点的二叉树。
7. 图的存储结构包括 数组 (邻接矩阵)、邻接表、十字链表 和 邻接多重表 等几种。图的遍历路径包括 深度优先遍历 和 广度优先遍历。
8. 常用的构造哈希函数的方法有 直接定址法、数字分析法、平方取中法、折叠法、除留余数法 和 随机数法。

### 十一、 选择题 (20 分)

请在你认为正确的答案所对应的字母上画“√”。

6. 在 C 语言中, 要存储一个 8 个字符的字符串, 至少需要声明大小为多少的一维字符数组? C  
 (A) 7 (B) 8 (C) 9 (D) 10

7. 两个矩阵  $A: m \times n$ ,  $B: n \times p$  相乘, 其时间复杂度为: B  
 (A)  $O(n)$  (B)  $O(mnp)$  (C)  $O(n^2)$  (D)  $O(n^3)$

8. 下列程序为将一条数据插入栈上:

```
void add(int top, element item)
{ if (top >= MAX_STACK_SIZE-1)
    return stack_full();
  stack[____]=item; }
```

则在  $stack[_____]$  的中括号内横线上的正确内容应为: A

- (A)  $++*top$  (B)  $*top++$  (C)  $*top--$  (D)  $*top$

9. 有如下函数:

```
void fun(struct node h1, struct node h2)
{ struct node *t;
  t=h1;
  while(t->next!='\0')
    t=t->next;
  t->next=h2;
}
```

其中形参  $h1$  和  $h2$  分别指向 2 个不同链表的第一个结点, 此函数的功能是: A

- (E) 将链表  $h2$  接到链表  $h1$  后  
 (F) 将链表  $h1$  接到链表  $h2$  后



- (G) 找到链表 h1 的最后一个结点由指针返回  
(H) 将链表 h1 拆分成两个链表

10. 一个栈的入栈序列是 abcde, 则栈的不可能输出序列是: C

- (A) edcba (B) decba  
(C) dceab (D) abcde

十二、 回答问题, 并给出理由。(10 分)

3. 设在一个有关串的程序编码当中, 有如下定义与赋值:

const char A[]={'a','b','c','\0'};

char B[]={'a','b','c','d','\0'};

... ..

for(i=0;i<4;i++)

{

A[i]='a';

B[i]='b';

}

... ..

在该程序编码中是否有错? 为什么?

参考答案:

有错。错在: A[i]='a';

A 定义的是串常量。一旦定义并赋值后, 不能再赋值。

4. 若 A 为一下三角矩阵数组, 则采用以行为主和以列为主的数据存放方式哪一种更合适? 为什么?

参考答案:

以行为主更合适。因为:

以行为主, A(i,j)存储于 B(k), 则  $k=[i(i-1)/2]+j$ 。

以列为主, A(i,j)存储于 B(k), 则  $k=[n(j-1)]-j(j-1)/2+i$ 。

可见以行为主的方式存储较简单。

十三、 根据要求编写算法。(20 分)

3. 线性表 A 和 B 均是按元素值递增有序排列, 均以单链表作存储结构。请编写一算法将表 A 和表 B 归并成一个按元素值递减有序排列的线性表 C (允许表中含有值相同的元素), 并要求利用原表空间。

参考答案:

void getUnionList(SqList &A,SqList &B,SqList &C){

node pa, pb, pc, q;

pa=A->next;

pb=B->next;

C=A;

A->next=null;

```

While( pa!=null & pb!=null){
    if (pa->data<=pb->data){
        q=pa;
        pa=pa->next;
        q->next=C->next;
        C->next=q;
    }
    else{
        q=pb ;
        pb=pb->next;
        q->next=C->next;
        C->next=q;
    }
}

```

```

if (pa!=null){
    while(pa!=null){
        q=pa ;
        pa=pa->next;
        q->next=C->next;
        C->next=q;
    }
}
if (pb!=null){
    while(pb!=null){
        q=pb ;
        pb=pb->next;
        q->next=C->next;
        C->next=q;
    }
}
}

```

4. 编写一个算法，对于输入的十进制非负整数，将它的八进制表示打印出来。

参考答案：

```

void print_oct(int dec_number){
    PSeqStack pastack;
    int temp=dec_number;
    if (temp<0){
        printf("Error!\n");
        return;
    }
    pastack=createEmptyStack_seq();
    if (pastack==NULL) return;
    while(temp>0){

```

```

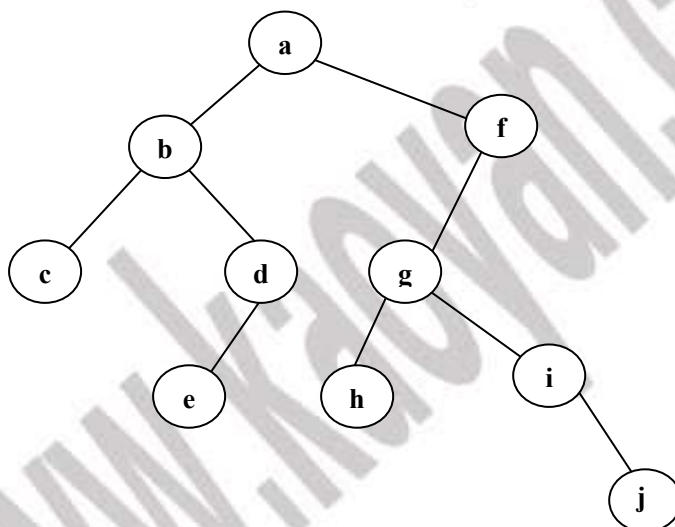
    push_seq(pastack,temp%8);
    temp/=8;
}
while(!isEmptySeq(pastack)){
    printf("%d",top_seq(pastack));
    pop_seq(pastack);
}
free(pastack);
}

```

十四、 回答以下问题，并给出计算或推理过程。（20 分）

3. 已知一棵二叉树的中序序列为 cbedahgijf，后序序列为 cedbhjigfa。给出其先序序列，并画出该二叉树。

参考答案：

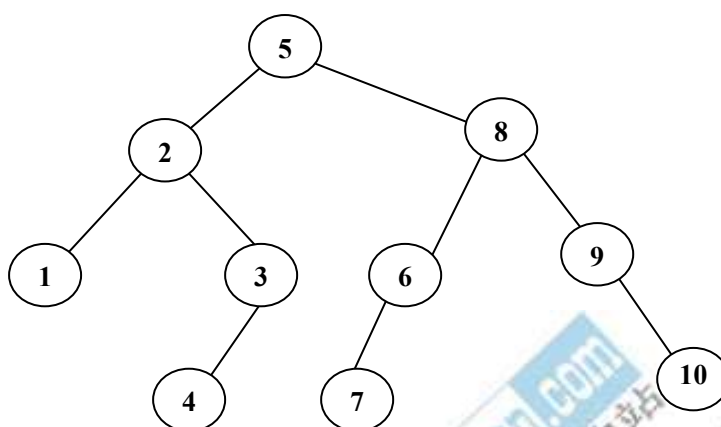


先序序列为 abcdefghij。

4. 画出对长度为 10 的有序表进行折半查找的一棵判定树，并求其等概率时查找成功的平均查找长度。

参考答案：

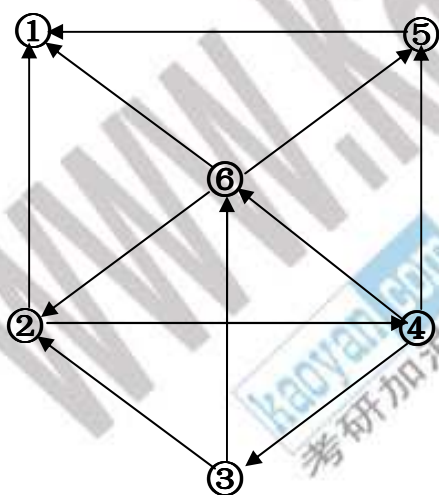




等概率时查找成功的平均查找长度为：  
 $1/10 (1*1+2*2+3*4+4*3) = 29/10 = 2.9$

十五、 已知如图所示的有向图，请给出该图的：

- (1) 每个顶点的入/出度；
- (2) 邻接矩阵；
- (3) 邻接表；
- (4) 逆邻接表；
- (5) 强连通分量。(10 分)



参考答案：

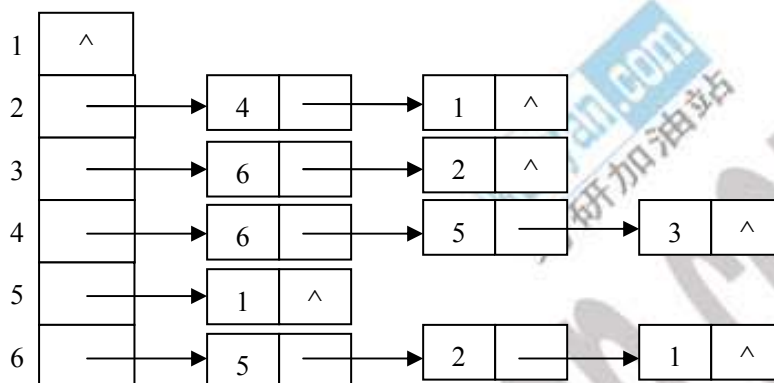
(1) 各个顶点的入/出度：

顶点	1	2	3	4	5	6
入度	3	2	1	1	2	2
出度	0	2	2	3	1	3

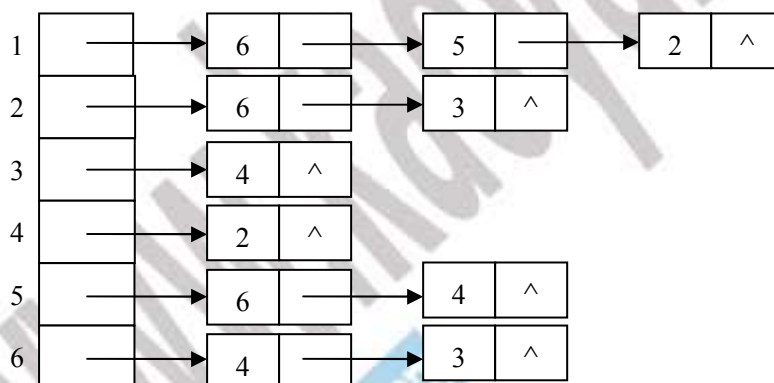
(2) 邻接矩阵

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

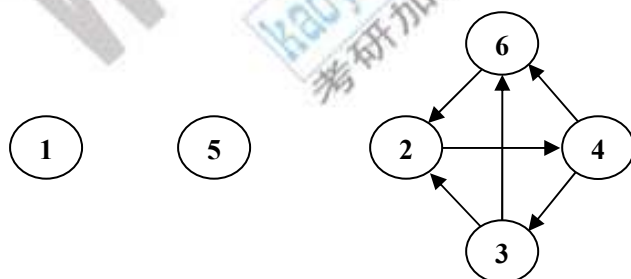
(3) 邻接表



(4) 逆邻接表



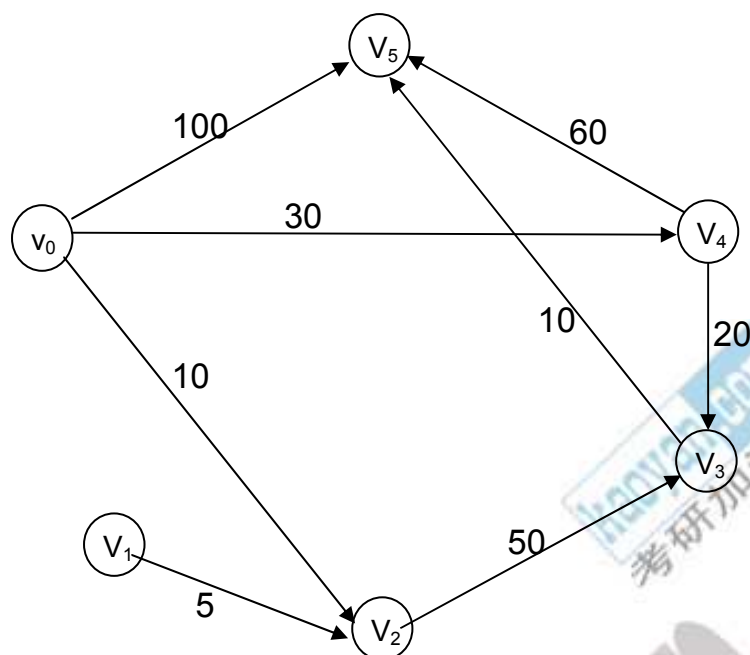
(5) 有 3 个强连通分量



十六、 下图是一个有向图，其中每条弧段上的数字表示该弧段的权值。

3. 请用 Dijkstra 算法计算  $v_0$  到各点的最短路径（要求给出计算过程）。

4. 给出用 C 语言描述的 Dijkstra 算法。(20 分)



参考答案:

最短路径:

始点	终点	最短路径	路径长度
v0	v1	无	
	v2	(v0,v2)	10
	v3	(v0,v4,v3)	50
	v4	(v0,v4)	30
	v5	(v0,v4,v3,v5)	60

用 C 语言描述的 Dijkstra 算法:

```

void ShorttestPath_DIJ(MGraph G, int v0, PathMatrix &P, ShortPathTable &D){
    for(v=0; v<G.vexnum; ++v){
        final[v]=FALSE; D[v]=G.arcs[v0][v];
        for(w=0; w<G.vexnum; ++w) P[v][w]=FALSE;
        if(D[v]<INFINITY) {p[v][v0]=TRUE; P[v][v]=TRUE;
    }
    D[v0]=0; final[v0]=TRUE;
    for(i=1; i<G.vexnum; ++i){
        min=INFINITY;
        for(w=0; w<G.vexnum; ++w)
            if(!final[w])
                if(D[w]<min){v=w; min=D[w];}
        final[v]=TRUE;
        for(w=0; w<G.vexnum; ++w)
            if(!final[w]&&(min+G.arcs[v][w]<D[w])){

```

```

        D[w]= min + G.arcs[v][w];
        P[w]=P[v]; P[w][w]=TRUE;
    }
}
}

```

十七、 请回答以下有关 C++语言的问题。(16 分)

5. 请比较一下值调用与引用调用的相同点和不同点。

参考答案:

值调用是指当发生函数调用时,给形参分配内存空间,并用实参来初始化形参(直接将实参的值传递给形参)。这一过程是参数值的单向传递过程,一旦形参获得了值,便与实参脱离了关系,此后无论形参发生了怎样的改变,都不会影响到实参。

引用调用将引用作为形参,在执行主调函数中的调用语句时,系统自动用实参来初始化形参。这样形参就成为实参的一个别名,对形参的任何操作也就直接作用于实参。

6. 什么叫作抽象类? 抽象类有何作用? 抽象类的派生类是否一定要给出纯虚函数的实现?

参考答案:

带有纯虚函数的类是抽象类。抽象类的主要作用是通过它为一个类族建立一个公共的接口,使它们能够更有效地发挥多态特性。抽象类的派生类不一定要给出纯虚函数的实现。

7. 什么叫作指针? 指针中存储的地址和这个地址中的值有何区别?

参考答案:

指针是一种数据类型,具有指针类型的变量称为指针变量。指针变量存放的是另外一个对象的地址,这个地址的值就是另一个对象的内容。

8. 什么叫拷贝构造函数? 拷贝构造函数何时被调用?

参考答案:

拷贝构造函数是一种特殊的构造函数,其形参是本类的对象的引用,其作用是使用一个已经存在的对象,去初始化一个新的同类的对象。拷贝构造函数在以下三种情况下会被调用:当用类的一个对象去初始化该类的另一个对象时;如果函数的形参是类对象,调用函数进行形参和实参结合时;如果函数的返回值是类对象,函数调用完成返回时。

十八、 建立基类 **Building**, 用来存储一座楼房的层数、房间数以及它的总平方英尺数。建立派生类 **Housing**, 继承 **Building**, 并存储下面的内容: 卧室和浴室的数量。另外, 建立派生类 **OfficeBuilding**, 继承 **Building**, 并存储灭火器和电话的数目。

然后, 用 C++语言编制应用程序, 建立住宅楼对象和办公楼对象, 并输出它们的有关数据。(14 分)

参考答案:

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

```
class Building
```

```

{
public:
    Building(int f, int r, double ft)
    {
        floors=f;
        rooms=r;
        footage=ft;
    }
    void Show()
    {
        cout << "    floors:"<<floors<<endl;
        cout << "    rooms:"<<rooms<<endl;
        cout << "    total area:"<<footage <<endl;
    }
protected:
    int floors;
    int rooms;
    int footage;
};

class Housing:public Building
{
public:
    Housing(int f, int r,double ft,int bd,int bth):Building(f,r,ft)
    {
        bedrooms=bd;
        bathrooms=bth;
    }
    void Show()
    {
        cout <<"\n HOUSING:\n";
        Building::Show();
        cout << "    Bedrooms:"<<bedrooms<<endl;
        cout << "    Bathrooms:"<<bathrooms<<endl;
    }
private:
    int bedrooms;
    int bathrooms;
};

class OfficeBuilding:public Building
{
    OfficeBuilding(int f,int r,double ft,int ph,int ex)
    {

```



```

        phones=ph;
        extinguishers=ex;
    }
    void Show()
    {
        cout <<"\n OFFICEBUILDING:\n";
        Building::Show();
        cout <<"  Phones:"<<phones<<endl;
        cout <<"  Extinguishers:"<<extinguishers<<endl;
    }
private:
    int phones;
    int extinguishers;
};

void main()
{
    Housing hob(5,7,140,2,2);
    OfficeBuilding oob(8,12,500,12,2);

    hob.Show();
    oob.Show();
}

```