

希赛网, 专注于**软考**、**PMP**、**通信考试**的专业 IT 知识库和在线教育平台。希赛网在线题库, 提供历年考试真题、模拟试题、章节练习、知识点练习、错题本练习等在线做题服务, 更有能力评估报告, 让你告别盲目做题, 针对性地攻破自己的薄弱点, 更高效的备考。

希赛网官网: <http://www.educity.cn/>

希赛网软件水平考试网: <http://www.educity.cn/rk/>

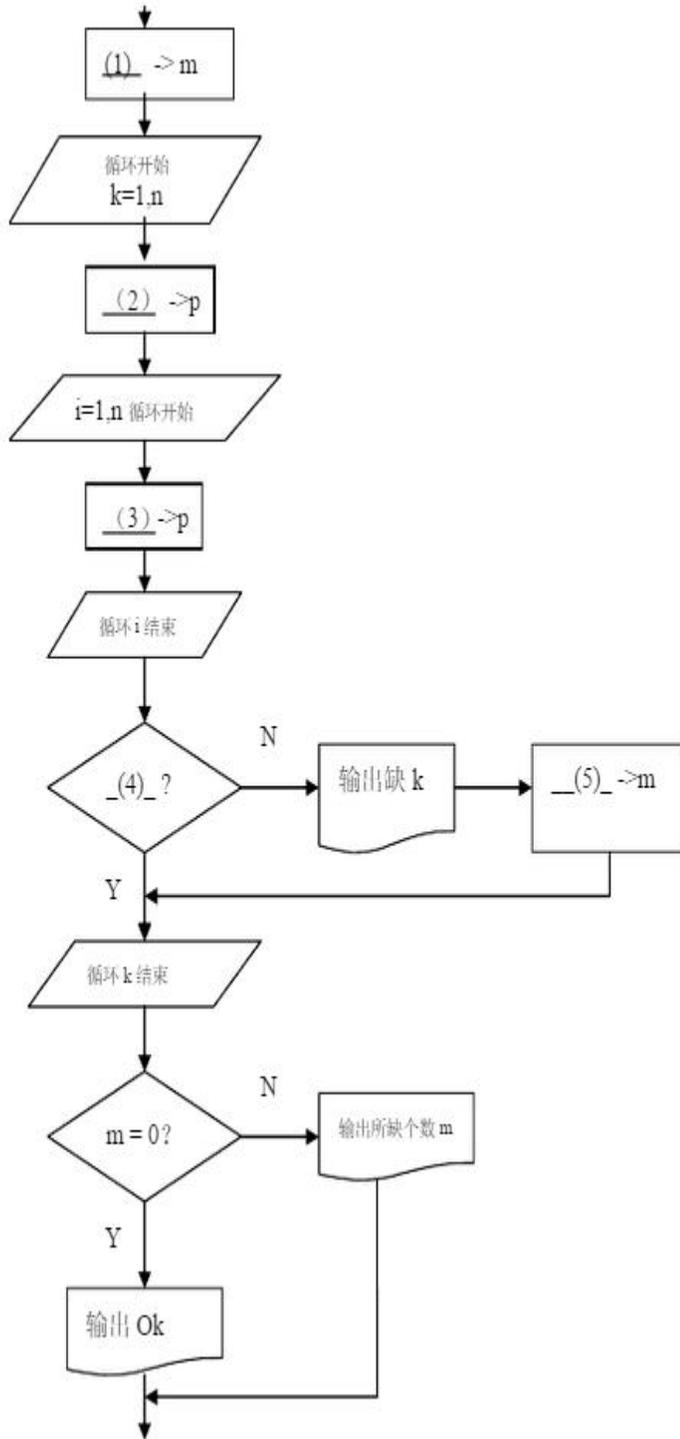
希赛网在线题库: <http://www.educity.cn/tiku/>

2011 下半年程序员案例分析真题答案与解析: <http://www.educity.cn/tiku/tp19319.html>

## 2011 年下半年程序员考试下午真题 (参考答案)

● 试题一 (共 15 分) 阅读以下说明和流程图, 填补流程图中的空缺 (1) ~ (5), 将解答填入答题纸的对应栏内。 【说明】 下图所示的流程图用于检查数组  $A[1:n]$  中的元素是否为自然数  $1\sim n$  的一个排序 (含有  $1\sim n$  各数), 若是, 则输出 OK, 否则输出所缺的自然数及其个数  $m$ 。 为检查  $A[1:n]$  中是否含有  $k$ , 只要判断  $P(k)=(A(1)-k)*(A(2)-k)*\dots*(A(n)-k)$  是否等于 0 即可。

【流程图】



- 试题二 (共 15 分) 阅读以下说明和 C 程序代码, 将解答写在答题纸的对应栏内。【说明】 下面是一个待修改的 C 程序, 其应该完成的功能是: 对于输入的一个整数 num, 计算机其位数 k, 然后将其各位数字逆序转换为字符串保存并输出。若 num 为负整数, 则输出字符串应有前缀“-”。例如, 将该程序修改正确后, 运行时若输入“14251”, 则输出“15241”; 若输入“-6319870”, 则输出“-0789136”。 下面给出的 C 程序代码中有五处错误, 请指出错误代码

所在的行号并给出修改正确后的完整代码行。

【C 程序代码】

行号	代码
1	#include<stdio.h>
2	#include<stdlib.h>
3	int main()
4	{
5	long num=0,t=0; char *pstr,i=0,k=0;
6	
7	scanf("%ld",&num); /*输入一个整数, 存入 num*/
8	t=num; k=num!=0?0:1;
9	while(t>=0){ /*计算位数*/
10	t=t/10;
11	k++;
12	}
13	
14	pstr=(char *) malloc((k+2)+sizeof(char));
15	If(pstr=0) return -1;
16	
17	i=0;
18	if(num<0){
19	num=-num
20	pstr[0]='-';i=1;
21	}
22	for(;k>0;k--){ /*形成字符串*/
23	pstr[i++]=num%10;
24	num=num/10;
25	}
26	pstr[k]='\0'; /*设置字符串结尾*/
27	printf("%s\n",pstr);
28	free(pstr);
29	return 0;
30	}

- 试题三 (共 15 分) 阅读以下说明和 C 代码, 填补 C 代码中的空缺 (1) ~ (5), 将解答写在答题纸的对应栏内。 【说明】 某市在进行市长选举, 该市共分为 M 个选区 (1<M<6, 从 1 开始顺序编号), 共有 N 个候选者参选 (1<N<5, 从 A 开始顺序编号)。经

投票后, 所得选票数据如下表所

选区	竞选者 A	竞选者 B	竞选者 C	竞选者 D
1	192	48	206	37
2	147	90	312	21
3	186	12	121	38
4	114	21	408	39
5	267	13	382	29

示。 现已将选票数据存入文本文件 data.txt, 该文件内容格式如下: 共有 M 行, 每行 N+1 个整数, 第一个整数为选区编号, 随后为 N 个候选者在该选区所得票数。 下面的程序先从文件中读入选票数据存入二维数组 data, 然后完成下列功能: a. 计算并显示每个竞选者获得的选票数及占总选票数的百分比; b. 如果任一竞选者获得的选票数超过总票数的 50%, 则显示竞选者获胜; c. 如果没有竞选者获得总选票数的 50% 以上, 则显示两位得票最高的竞选者需再进行决选。 在下面的程序代码中, 竞选者 A 的编号为 1, 竞选者 B 的编号为 2, 以此类推。

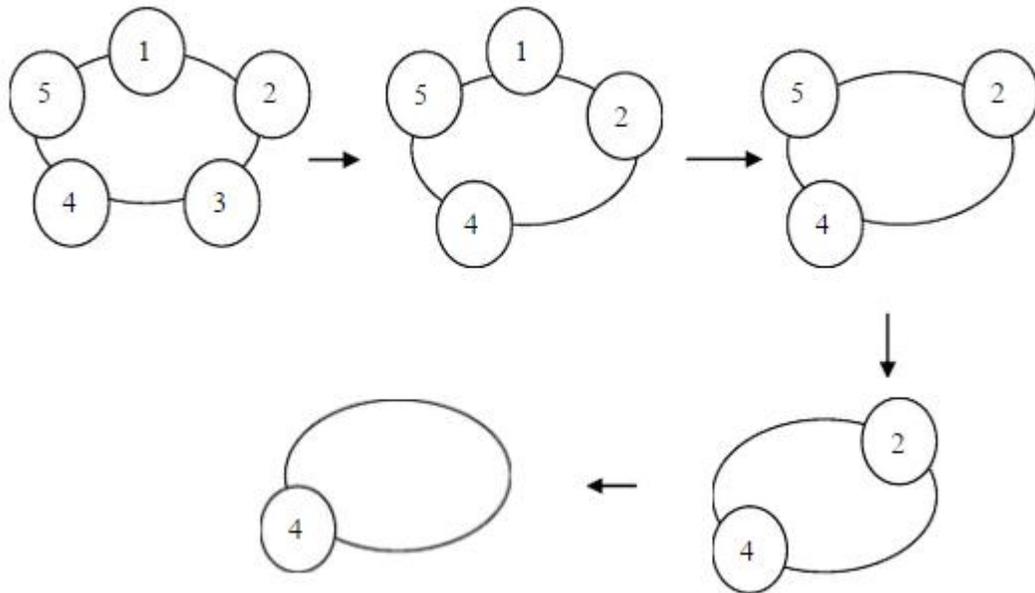
【C 代码】

```
#include<stdio.h>
#include<stdlib.h>
#define M 5 /*选区数*/
#define N 4 /*竞选者人数*/
int main(int argc, char argv[])
{
    FILE *fin;
    int data[M+1][N+1]; /*data[][0]存放选区编号, data[M][J]存放 J 号竞选者得票总数*/
    int maxp1,cd1; /*maxp1、cd1 存放得票最多者的票数和竞选者编号*/
    int maxp2,cd2; /*maxp2、cd2 存放得票次多者的票数和竞选者编号*/
    int i,j, totalP; /*totalP 存放总票数*/
    fin=fopen("dat
(3) A. txt","r");
    if(!fin)return -1;
    i=0; j=0;
    While(!feof(fin)){/* 从文件读入数据 */
    Fscanf(fin,"%d",&data[i][j]);
    __ (1) __;
    if(j>N) {j=0;i++;}
    }
    fclose(fin);
    totalP =0;maxp1=0;maxp2=0;cd1=0;cd2=0;
    for(j=1;j<N+1;j++)
    {
        data[M][j]=__ (2) __; /* 设置 j 号竞选者的票总数初始值 */
        for(i=0;i<M; i++)
            data[M][j]+=data[i][j];
        if(data[M][j]>MAXP1)
        {
            __ (3) __; cd2=cd1;
            MAXP1=data[M][j]; cd1=j;
        }
    }
}
```

```

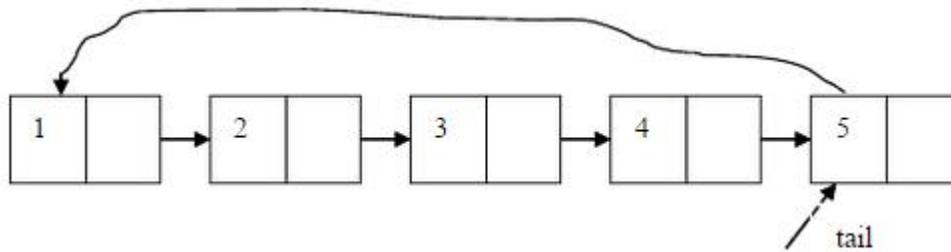
else
    if(data[M][j]> MAXP2){ MAXP2=data[M][j];cd2=j;}
    __ (4) __; /* 计算总票数 */
}
for(j=1;j<N+1;j++)
{
    printf("candidate—%c:%d ",j+'A'-1,data[M][j]);
    printf("%.2lf\n",(double)data[M][j]/totalP);
}
printf("\n");
if(__ (5) __>0.5) /* 判断得票数最多者的得票率 */
printf("winner:candidate-%c\n",cd1+'A'-1);
else
printf("result : candidate-%c pk candidate-%c\n",cd1+'A'-1,cd2+'A'-1);
system("pause");
return 0;
}
    
```

● 试题四 (共 15 分) 阅读以下说明和 c 函数, 填补 c 函数中的空缺 (1) — (5), 将解答写在答题纸的对应栏内。 【说明】 约瑟夫问题如下所述: 有 n 个人 (编号为 1~n) 围成一圈, 从第一个人开始, 按照顺时针方向从 1 开始计数到 m (即数到第 m 个人), 让其出圈, 然后再从其顺时针方向的下一个人开始, 依次计数到 m 并让其出圈, 重复这个过程, 直到所有人都出圈, 试给出出圈者的顺序。 以 n=5,m=3 为例, 其中圈顺序为 3,1,5,2,4, 过程如下图所示



示。

下面的函数 Joseph\_\_ (4) \_\_ 在单向循环链表表示的基础上诉出圈的过程。 n 为 5 时的单向循环链表结构如下图所示



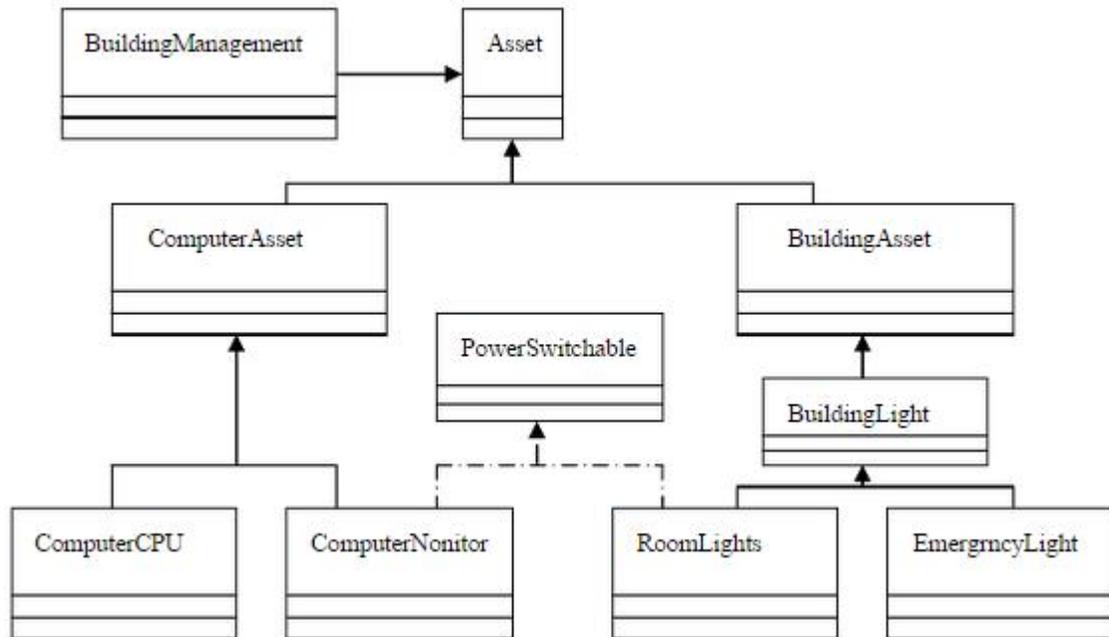
示。

表的结合类型定义如下： `Typedef struct Node int no; struct Node*next; }node,*LinkList;` 函数 Joseph (LinkList tail,int n,int m) 的处理思路如下： (1) 用 k 计数，每次都从 0 开始，当计数到 m-1 时结束本次计数； (2) 通过指针 p 查找出圈者所对应的结点，当 K 的值等于是 m-1 时，P 应指向出圈者对应结点的前驱结点； (3) 通过删除结点表示出圈处理； (4) 当 m 大于圈中剩余人数时，为了避免重复计数，用模运算修改 m 的值； (5) 计数和删除操作完成后再恢复 m 的原值；

#### 【C 函数】

```
void Joseph (LinkList tail, int n, int m)
{
    LinkList p,q;
    int k,i,old_m=m;
    p=tail;
    for(i=n;i>1;i--) /*i 表示圈中剩余人数*/
    {
        m=m%i; /*避免重复计数*/
        if(0==m) m=__ (1) __;
        k=0;
        while(k<__ (2) __)
        { __ (3) __; k++; }
        printf("%d\n",__ (4) __); /*输出出圈者的编号*/
        q=p->next;
        __ (5) __=q->next; /*删除出圈者对应的结点*/
        free(q);
        M=old_m;
    }
    printf("%d\n",p->No);
}
```

- 试题五 (共 15 分) 阅读以下说明、图和 C++ 代码，填补 C++ 代码中的空缺 (1) — (5)，将解答写在答题纸的对应栏内。 【说明】 已知某公司主要有两大类耗电资产 (Asset)：计算机 (ComputerAsset) 和建筑物 (BuildingAsset)。为了节约能源，通过控制各种电源，将可关闭的房灯、计算机显示器等在夜间关闭。 为了实现上述需求，设计了如下图所示的类图，并用下面的 C++ 代码加以实现。



【C++代码】

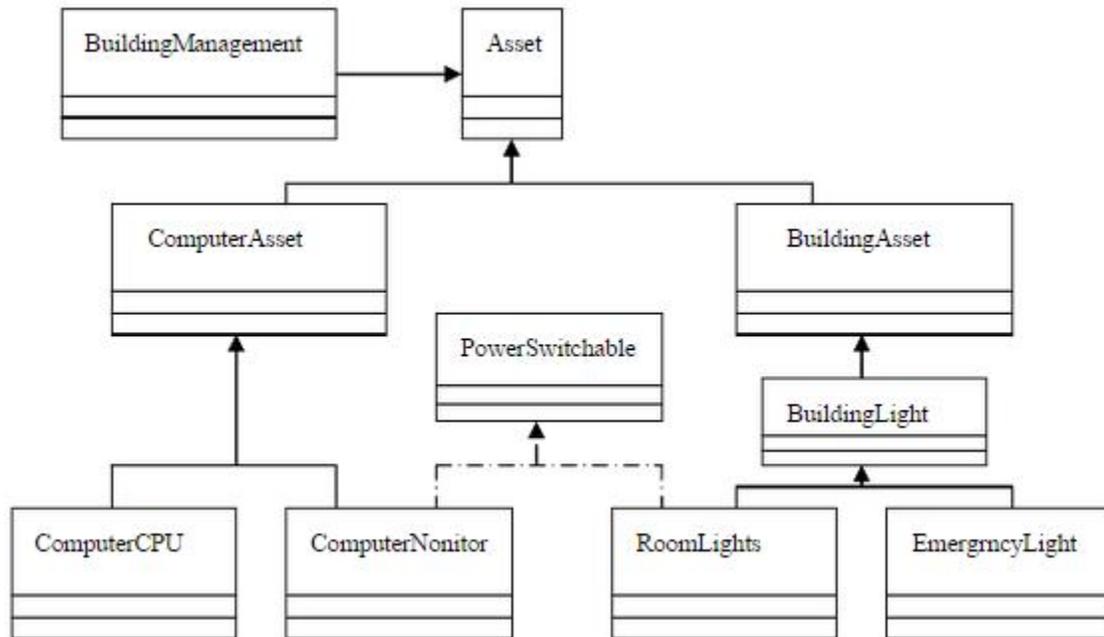
```

#include<typeinfo>
#include<iostream>
using namespace std;
class Asset{ /*通用资产，基类*/
public: virtual ~Asset(){};
};
class PowerSwitchable{ /*抽象基类，可在夜间关闭电源的物体接口*/
public: virtual void powerDown()=0; /*powerDown()函数接口*/
virtual void powerUP()=0; /*powerUp()函数接口*/
};
class computerAsset: public Asset{ /*计算机资产*/
protected: int deskNumber;
public:
computerAsset(int desNumber){ this->deskNumber= deskNumber;}
};
class computerCPU __ (1) __ /*计算机主机，永不关闭*/
public:
computerCPU(int desNumber): ComputerAsset (desNumber){ }
};
class computerMonitor __ (2) __{ /*计算机显示器*/
public:
computerMonitor(int roomNumber):omputerAsset(roomNumber),PowerSwitchable(){}
~computerMonitor (){}
void powerDown(){...../*关电源，代码省*/}
void powerUp(){...../*开电源，代码省*/}
};
/*BuildingAsset、 BuildingLight
、 EmergencyLight 和 RoomLigths 代码省*/
class BuldingManagement{
private:

```

```
Asset* things[24]; int numItems;
public:
void goodnight(){ /*值班员定时“关闭”时调用, 关闭可关闭的电源*/
for(int i=0;i<numItems;i++){
__ (3) __ ps=dynamic_cast<powerSwitchable*>(things[i]);
if(ps!=0)
ps->powerDown();
}
}
/*goodMorning()与 goodNight()类似, 依次调用 powerUp(),实现省*/
void add(Asset*thing){ /*为建筑添加资产*/
yhings[__ (4) __]=thing;
}
};
int main(){
BuildingManagement* bl=__ (5) __ BuildingManagement();
bl->add(new RoomLights(101)); //101 房间的控制灯
bl->add(new EmergencyLight(101)); //101 房间的应急灯
bl->add(new ComputerCPU(10104)); //101 房间 4 号桌上的计算机
bl->add(new ComputerMonitor(10104)); //101 房间 4 号桌上的计算机显示器
bl->goodNight();
Delete bl;
}
```

- 试题六 (共 15 分) 阅读以下说明、图和 Java 代码, 填补 Java 代码中的空缺 (1) — (5), 将解答写在答题纸的对应栏内。【说明】 已知某公司主要有两大类耗电资产 (Asset): 计算机 (ComputerAsset) 和建筑物 (BuildingAsset)。为了节约能源, 通过控制各种电源, 将可关闭的房灯、计算机显示器等在夜间关闭。 为了实现上述需求, 设计了如下图所示的类图, 并用下面的 Java 代码加以实现。



**【Java 代码】**

```

abstract class Asset{          /*通用资产，基类*/
interface PowerSwitchable{    /*可在夜间关闭的电源的物体实现该接口*/
    public void powerDown();
    public void powerUP();
}
abstract class BuildingAsset extends Asset{ /*建筑物资产*/
    protected int room;
    public BuildingAsset(int room){ this.room=room; }
}
abstract class BuildingLight extends BuildingAsset{ //灯的通用信息：fluorescent /
    incandescent 等，略
    BuildingLight(int room Number){ super(roomNumber); }
}
class EmergencyLight __ (1) __{ //应急灯，永不关闭
    EmergencyLight(int roomNumber){
        Super(roomNumber);
    }
}
class RoomLights __ (2) __{
    RoomLights(int roomNurober){ super(roomNumber);}
    Public void powerDown(){ //关电源，代码省 }
    Public void powerUp(){ //开电源，代码省 }
}
/*ComputerAsset、ComputerCPU 和 ComputerMonitor 代码省*/
public class BuildingManagement{
    Asset things[]=new Asset[24];
    int numItems=0;
    public void goodnight(){ //值班员定时“关闭”时调用，关闭可关闭的电源
        for(int i=0;i<things.length;i++)
            if(things[i] instanceof __ (3) __)
    
```

```
        ((powerSwitchable)things[i]).powerDown();
    }
    /*goodMoming()与 goodNight()类似, 依次调用 powetUp(),其实现细节此处省*/
    public void add(Asset thing){ /*为建筑添加资产*/
        Things[__(4) __]=thing;
    }
    public static void main(String[]args){
        BuildingManagement* bl=__(5) __ BuildingManagement();
        bl.add(new RoomLights(101)); //101 房间的控制灯
        bl.add(new EmergencyLight(101)); //101 房间的应急灯
        bl.add(new ComputerCPU(10104)); //101 房间 4 号桌上的计算机
        bl.add(new ComputerMonitor(10104)); //101 房间 4 号桌上的计算机显示器
        bl.goodNight();
        delete bl;
    }
}
```