

西南交通大学 2021 年硕士研究生

招生入学考试试卷

试题代码:840

试题名称:数据结构与程序设计

考生请注意:

- 1.本试题共四题,共 12 页,满分 150 分,请认真检查;
- 2.答题时,请直接将答题内容写在考场提供的答题纸上,答在试卷上的内容无效;3.请在答题纸上按要求填写试题代码和试题名称;
- 4.试卷不得拆开,否则遗失后果自负。

一、单项选择题 (共 52 分, 每小题 2 分, 共 26 小题)(答在试卷上的内容无效)

- 1.一个算法应该是
A.程序
B.问题求解的步骤
C.数据结构+程序
D.以上都不对
2. 以下术语与存储结构无关的是
A.循环队列 B.双向链表 C.哈希表 D.堆栈
3. n 元顺序表在第 i 个元素之前插入一个元素的算法时间复杂度为
A. $O(1)$ B. $O(n)$ C. $O(n\log_2 n)$ D. $O(n^2)$
4. 对一个头指针为 `head` 的带附加头结点的循环单向链表, 若结点指针域名为 `next`, 则判定该表为空表的逻辑表达式是
A.`head==NULL`
B. `head!=NULL`
C. `head->next==NULL`
D. `head->next==head`
5. 若一个堆栈的输入元素序列为 $1,2,3,4,\dots,n-1,n$, 若输出序列的第 1 个元素是 n , 则输出序列的第 $i(1 < i \leq n)$ 个元素是
A. $n-i+1$
B. $n-i$
C. i
D.不能确定
- 6.若用一个长度为 6 的一维数组(元素下标范围 0~5)实现循环队列, 若队头下标 `front` 指向队头元素之前空白元素位置, 队尾下标 `rear` 指向队尾元素位置, 元素入队方向为下标加 1 方向, 则当 `rear,front` 的值分别为 0 和 3 时, 若从队列中删除一个元素再插入两个元素后, `rear` 和 `front` 的值分别为
A.1 和 5 B.4 和 2 C.2 和 4 D.5 和 1
- 7.若一棵二叉树具有 10 个度为 2 的结点, 则度为 0 的结点个数是

A.8 B.9 C.10 D.11

8. 一个具有 1025 个结点的二叉树的高度 h 为

A.11 B.10 C.11 至 1025 之间 D.10 至 1024 之间

9. 二叉树的先序遍历序列为 EFHIGJK，中序遍历序列为 HFIEJKG，则该二叉树树结点的右子树的根是

A.E B.F C.G D.H

10. 有 n 个顶点的连通无向图，其边的个数至少为。

A. $n-1$ B. n C. $n+1$ D. $n(n-1)/2$

11. 下列关于 AOE 网的叙述中，不正确的是

- A. 关键活动不按期完成就会影响整个工程的完成时间
- B. 所有关键活动提前完成，那么整个工程才会提前完成
- C. 非关键活动不按期完成，不一定影响整个工程的完成时间
- D. 任何一个关键活动提前完成，那么整个工程将会提前完成

12. 具有 n 个关键字的二叉排序树各关键字查找概率相同，则树形在最糟糕的情况下成功查找的平均查找长度为

A. $(n+1)/2$ B. n^2 C. $\log_2 n$ D. $n \log_2 n$

13. 若需要在 $O(n \log_2 n)$ 时间完成对数组元素的排序，且要求排序是稳定的，则可选的排序方法是

A. 快速排序 B. 二路归并排序 C. 堆排序 D. 直接插入排序

14. C 语言中用于结构化程序设计的三种基本结构是

- A. for、while、do-while
- B. if、switch、break
- C. 顺序结构、选择结构、循环结构
- D. if、for、continue

15. C 语言中，凡未指定存储类别的局部变量其隐含存储类别是_(15)_。

- A. auto
- B. static
- C. extern
- D. register

16. 当 A 的值为奇数时，表达式的值为"真"， A 的值为偶数时，表达式的值为"假"。则以下不能满足要求的表达式是

- A. $A \% 2 == 1$
- B. $!(A \% 2 == 0)$
- C. $A \% 2$
- D. $!(A \% 2)$

17. 以下对 C 语言中共用体类型数据的叙述正确的是

- A. 可以对共用体变量名直接赋值

- B.一个共用体变量中可以同时存放其所有成员
- C.一个共用体变量中在每一瞬时只能存放其中一个成员
- D. 共用体类型定义中不能出现结构体类型的成员

18. 对两个数组 a 和 b 进行如下初始化:

```
char a[ ]="ABCDE";
char b[]={ 'A','B','C','D','E'};
```

则以下叙述正确的是

- A. a 数组比 b 数组长度长
- B. a 与 b 长度相同
- C. a 和 b 中都存放字符串
- D.a 与 b 数组完全相同

19. 已知:char *language[]={ "Fortran","Basic","Pascal","Java","C"};则 language[1]的值是

- A.一个字符
- B.一个地址
- C.一个字符串
- D.一个不定值

20. 若有定义: int a[2][3];则对 a 数组的第 i 行第 j 列 (假设 i,j 已正确说明并赋值) 元素值的正确引用为_(20)

- A.*(*(a+i)+j)
- B. (a+i)[j]
- C.*(a+i+j)
- D.*(a+i)+j

21. 阅读下面的程序

```
#include<stdio.h>
void main()
{
    int x,y;
    scanf("%2d%3d",&x,&y);
    printf("x=%d,y=%d\n", x,y);
}
```

如果从键盘上输入 123456<回车>, 则程序的运行结果是

- A. x=12,y=3456
- B. x=12,y=345
- C. x=123456,y=0
- D.输入有误

22.以下程序的输出结果是

```
#include<stdio.h>
typedef struct
{
    int b;
    int c;
} NAME;
void f(NAME *p)
{
```

```

    p->b+=1;
    p->c+=2;
}
void main()
{
    NAME a= {1,2};
    f(&a);
    printf("%d,%d",a.b,a.c);
}

```

A.2,2 B.1,2 C. 1,4 D.2,4

23. 以下程序的输出结果是

```

void main()
{
    char ch;
    ch= 'A'+'6'-'4';
    printf("%c\n",ch);
}

```

- A. A
- B. B
- C. C
- D. D

24.以下 scanf 函数调用语句中对结构体变量成员的不正确引用是

```

struct student
{
    char name[20];
    int age;
    int sex;
} stud[5],*p;
p=stud;

```

- A.scanf("%s",stud[0].name);
- B.scanf("%d",&stud[0].age);
- C.scanf("%d",&(p->sex));
- D.scanf("%d",p->age);

25.以下程序的输出结果是

```

#include<stdio.h>
#include<string.h>
void main()
{
    int len;

```

```

char*s="ab\n89\0mn";
len = strlen(s);
printf("len=%d\n", len);
}

```

A. len=2 B. len=5 C. len=6 D. len=10

25. 以下程序的输出结果是

```

void f(int *q, int n)
{
    for( int i=0; i<n; i++) (*q++)++;
}
void main()
{
    int a[5]={1,3,5,7,9},i;
    f(a,5);
    for(i=0; i<5; i++)
        printf("%3d", a[i]);
}

```

二、填空题(共 30 分，每空 1 分，共 30 空)(答在试卷上的内容无效)

- 抽象数据类型(ADT)定义为一个数学模型以及该模型之上的一组_____的总称。
- 数据的存储结构包括_____存储结构和链式存储结构与两类。
- 中缀表达式 $((a+b)/c-d)*e$ 对应的后缀式为_____
- 在拥有 421 个结点的完全二叉树中,叶子结点的数目是_____
- 8 行×8 列对称矩阵采用行序为主序顺序存储其下半三角(含主对角线)元素于一维数组,若矩阵行号、列号以及一维数组元素下标均从 0 开始,则矩阵 6 行 5 列元素在一维数组中的下标为_____
- 对 n 元序列进行快速排序,当初始序列已经有序时,递归所需参数栈的空间复杂度为__
- 有向图的三种常用存储结构分别是邻接矩阵_____、十字链表。
- 若 Huffman 二叉树有 n 个叶子结点,则其结点总数为_____
- 以下算法函数在带附加头结点的单向链表中查找 data 域值等于 x 的数据结点,若查找成功,返回该数据结点的地址,查找失败返回 NULL。已知各数据结点的 data 域值互不相同。函数需要实现以下要求:
每个结点中有一个 int 型域 freq,表示该结点被成功查找的次数。一次成功查找后,成功查找到的数据结点的 freq 域值需自加 1。链表要求根据 freq 值由大到小的次序连接各数据结点,使得访问频率高的结点离附加头结点更近。每次成功查找后,应使各数据结点仍然按 freq 域值由大到小连接。请填空,使算法完整。

```
typedef struct node
```

```

{
    ElemTp data;
    int freq;
    struct node *next;
} LNode, *LNPtr;
LNPtr locate(LNPtr h, ElemTp x)//h 为附加头结点指针
{
    LNPtr pr=h, ph->next, qr, q;
    while(p)
    {
        if(_____) break;
        pr=p;
        p=p->next; //pr 始终指向*p 的直接前驱
    }
    if(!p) return NULL;//查找失败,返回 NULL
    p->freq++;
    if(pr==h||pr->freq>=p->freq) return p; /*p 连接位置保持不变
    //以下将调整结点*p 到正确的位置
    /*p->freq 自加 1 前, 有 pr->freq>=p->freq 为真;自加 1 后, 若 pr->freq<p->freq 为真,
    此时必有 pr->freq 的值与 p->freq 自加 1 之前的值相等*/
    qr=h;
    q=h->next;
    while(q->freq_____pr->freq)//填关系运算符
    {
        qr=q;
        q=q->next;
        pr->next=_____ ;
        qr->next=p;
        p->next=q;
        return _____
    }
}

```

10.已知某二叉树根结点地址 bt, 以下算法函数填充每个结点的 int 型属性 num。要求 num 域值应为以该结点为根的子树中所有结点的数目。比如:某结点的左子树有 3 个结点, 右子树有 8 个结点, 则该结点的 num 域值应填充为 12, 表示该子树共有 12 个结点。请填空, 使算法完整。

```

typedef struct tnode
{
    int num;
    struct tnode *lchild, *rchild;
} BTreeNode, *BTPtr;
void fillnum(BTPtr bt)

```

```

{
    if(!bt) return;
    fillnum(bt->lchild);
    fillnum(_____);
    if(!bt->lchild&&!bt->rchild)//处理叶子结点
    {
        bt->num=_____;
        return;
    }
    if(bt->lchild&&!bt->rchild)
    {
        bt->num=bt->lchild->num+1;    //只有左子树
        return;
    }
    if(!bt->lchild&&bt->rchild)
    {
        bt->num=-bt->rchild->num+1;    //只有右子树
        return;
    }
    bt->num=_____;
}

```

11. 对于不需要带返回值的 C 语言函数，应当用关键字_____定义函数。
12. 若有 `int a=12,b=9,c=5; printf("%d", a&b|c);` 输出的结果是_____。
13. 以下程序先将字符串 t 连接到另一个字符串 s 的尾部，再输出连接后的字符串，请填空。

```

#include<stdio.h>
void main(
{
    int i=0,j=0;
    char s[80],t[30];
    gets(s);
    gets(t);
    while( s[i]!='\0') i++;
    while( (s[_____]=t[_____])!='\0');
    puts(s);
}

```

14. 输入 10 个用户的用户名和密码，用户名为 15 个字符以内的字符串，密码为长度 6 个字符的字符串。新建一个二进制输出文件，将输入的用户名和密码以结构体的形式存入文件，要求在存放密码时将密码中每个字符的 ASCII 码值减 1。

```

#include<stdio.h>
#include<stdio.h>
#include<stdlib.h>
struct user

```

```

{
    char name[16];/* name 数组存放用户名*/
    char code[7];/*code 数组存放用户密码*/
} use;
void main()
{
    int i,j;
    FILE *fp;
    char fname[20];
    scanf("%s", fname);
    if((fp=fopen(fname,_____))==NULL)
    {
        printf("File creat failed!\n");
        exit(0);
        for(i=0; i<10; i++)
        {
            scanf("%s %s", use.name, use.code);
            for(j=0; j<6; j++)
            _____
            fwrite(&use, sizeof(struct user),1, fp);
        }
        _____
    }
}

```

15.完善程序,使其能够按以下形式输出杨辉三角形。

```

1
1   1
1   2   1
1   3   3   1
1   4   6   4   1
.....

```

```

#include<stdio.h>
#define N 10
void main(
{
    int i, j, a[N][N];
    for(i=0; i<N; i++)
        a[i][0]=a[i][i]=1;
    for(i=2; i<N; i++)
        for(j=1; _____; j++)
            _____;
    for(i=0; i<N; i++)
    {

```



```

        for(j=0; _____; j++)
            printf("%5d", a[i][j]);
        printf("\n");
    }
}

```

16. 有以下程序段:

```

s= 1.0;
for(k=1; k<= n; k++)
    s=s+1.0/(k*(k+1));
printf("%f\n", s);

```

请填空,使下面的程序段的功能完全与之相同。

```

s=0.0;
_____;
k=0;
do
{
    s+=d;
    k++;
    d=1.0/ (k*(k+1));
}while(_____);
printf("%f\n", s);

```

17. 函数 min 的功能是:查找并返回带头结点的单链表中 data 数据域的最小值,请填空完成程序功能。

```

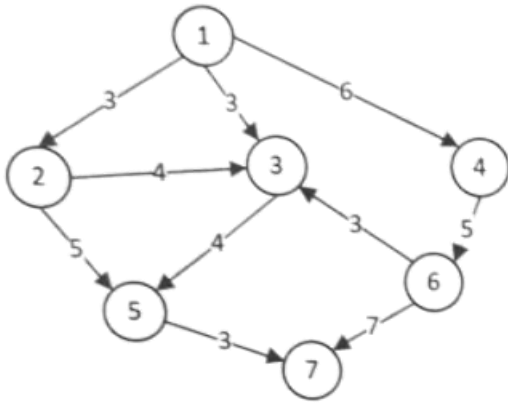
struct node
{
    int data;
    struct node *next;
};
int min(struct node *first)
{
    struct node *p;
    int m;
    _____
    m=p->data;
    p=p->next;
    for(; p !=NULL; _____)
        if(p->data <= m)
            _____;
    return m;
}

```

三、简答题(共 48 分,前 4 小题每小题 6 分,后 6 小题每小题 4 分)(答在试卷上的内容无效)

1. 某有向图如下图所示。(6 分)

(1)写出从 1 号顶点出发，深度优先遍历结点访问次序（假定邻接点的求取次序为顶点编号升序）。



(2)该图是否为有向无环图(DAG)?若是 DAG, 试写出一个顶点的拓扑有序序列;若不是 DAG, 请说明理由。

(3)若忽略所有边的方向(将该图视为一个无向图)，请画出一棵最小生成树(画图时，请保持各顶点相对位置与原图相同;生成树的每条边需要标注权值)。

2. 使用哈希函数 $H(K)=K \bmod 11$ ，把一个整数值转换成哈希表下标，现要把 8 个数据 1,13,30,12,38,22,27,41 依次插入到哈希表中(下标范围 $0 \cdots 10$)。(6 分)

(1)使用线性探测法构造哈希表

(2)若各数据元素查找概率相同，计算成功查找时的平均查找长度 $ASL_{成功}$ 。

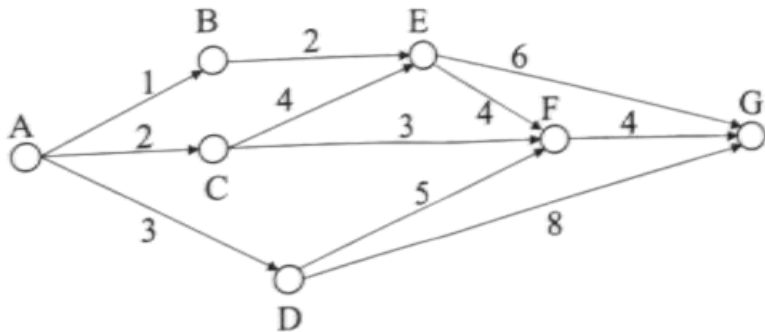
3.对 10 个关键字 45,20,15,63,8,30,6,72,12,54 进行由小到大排序。(6 分)

(1)以 45 为支点（枢轴），进行一趟快速排序，写出划分的结果。

(2)进行增量为 3 的一趟希尔排序，写出得到的序列。

(3)若进行堆排序,写出初始建堆(大根堆)后得到的序列。

4.AOE 网络如下图所示，请计算各顶点的最早开始时间和最晚开始时间，然后写出关键路径。(6 分)



5. 给出下面程序的运行结果（4 分）

```
#include<stdio.h>
void main()
```

```

{
    int a=5, b=4,c=6;
    c -=++b;
    b *=a+c;
    {
        int b= 3,c= 12;
        c/=b* 2;
        a-= c;
        printf("%d, %d, %d, ", a, b, c);
        a += --c;
    }
    printf("%d,%d,%d\n", a, b, c);
}

```

6. 给出下面程序的运行结果 (4 分)

```

#include<stdio.h>
#define PR(ar) printf("ar=%d ", ar)
void main()
{
    int j, a[]={1,2,3,4,5,6,7},*p=a+4;
    for (j=3; j; j--)
        switch(j)
        {
            case 1:
            case 2:
                PR(*(p++));
                break;
            case 3:
                PR(*(--p));
        }
}

```

7. 以下程序中函数 f 的功能是:当 flag 为非 0 时, 进行由小到大排序;当 flag 为 0 时, 进行由大到小排序, 给出下面程序的运行结果 (4 分)

```

#include<stdio.h>
void f( int b[], int n, int flag )
{
    int i,j, t;
    for ( i=0; i<n-1; i++)
        for (j=i+1; j<n; j++)
            if ( flag ? b[i]>b[j]:b[i]<b[j])
            {
                t=b[i];
                b[i]=b[j];
                b[j]=t;
            }
}

```

```

    }
}

void main()
{
    int a[10]= {8,7,10,9,6,1,5,3,4,2},i;
    f(&a[3],5,0);
    f(a,6,1);
    for(i=0; i<10; i++)
    {
        printf("%3d", a[i]);
    }
}

```

8. 给出下面程序的运行结果（4分）

```

#include <stdio.h>

struct s
{
    int n;
    int *m;
} *p;

int f[5]= {100,200,300,400,500};
struct s arr[5]= {10,&f[0], 20,&f[1], 30,&f[2], 40,&f[3],50,&f[4]};

void main()
{
    p=arr;
    printf("%d ",++p->n);
    printf("%d ",(++p)->n);
    printf("%d\n",++(*p->m));
}

```

9. 给出下面程序的运行结果（4分）

```

#include <stdio.h>

void main()
{
    int i;
    char a[]="Time", b[]="Tom";
    for(i=0; a[i]!='\0' && b[i]!='\0'; i++)
        if(a[i]==b[i])
            if(a[i]>= 'a'&& a[i]<='z')
                printf("%c",a[i]-32);
            else printf("%c",a[i]+32);
        else printf("***");
}

```

10. 给出下面程序的运行结果（4分）

```

#include <stdio.h>
void fun(int *a, int n)
{
    int i,j,t;
    for(i=0; i<n-1; i+=2)
        for(j=i+2; j<n; j+=2)
            if(a[i]<a[j])
            {
                t=a[i];
                a[i]=a[j];
                a[j]=t;
            }
}
void main()
{
    int b[10]= {1,2,3,4,5,6,7,8,9,10},i;
    fun( b,10);
    for(i=0;i<10; i++)
        printf("%3d",b[i]);
    printf("\n");
}

```

四、算法及程序设计题（共 20 分，每小题 10 分，共 2 小题)(答在试卷上的内容无效)

1. 已知二叉树采用二叉链表存储结构，结点及结点指针数据类型定义如下：

```
typedef struct node {int data; struct node *lchild, *rchild; } BTNode,*BT;
```

试写出 C++ 算法函数 void fmin(BT root, int &min); 求二叉树结点 data 域的最小值存于形参变量 min 中。其中，root 表示二叉树根结点地址。

也可以写出 C 语言函数 void fmin(BT root, int *pmin);

已知 int 型长度为 4 字节，则 int 型可以表示的最大整数为 2147483647。

调用时，int min=2147483647;

fmin(root, min); //C++ 语言

或 fmin(root, &min); //C 语言

调用结束后，变量 min 值为二叉树中所有结点 data 域的最小值。

2. 编写程序，定义一个一维整型数组，从键盘输入 10 个互不相同的整数，将数组中的最大值与数组中的第一个元素交换，数组中最小值与最后一个元素交换，输出交换后的数组元素。

要求：

(1) main() 函数的调用形式如下：

```

void main()
{
    int number[10];
    input(number, 10);
    max_min_change(number, 10);
    output(number, 10);
}

```

}

(2)要求数据输入格式为:4,5,1,6,7,9,10,8,3,2

数据的输出格式为:10,5,2,6,7,9,4,8,3,1

请编写相关函数，完成输入、查找并按要求交换最大和最小值、输出等功能。