

# 北京航空航天大学 2014 年

## 硕士研究生入学考试试题

科目代码: 991

### 数据结构与 C 语言程序设计 (共 8 页)

考生注意: 所有答题务必书写在考场提供的答题纸上, 写在本试题单上的答题一律无效 (本题单不参与阅卷)。

#### 一、填空 (本题共 20 分, 每小题各 2 分)

1. 设带有头结点的非空单向循环链表的链结点构造为 

data	link
------	------

, rear 为指向链尾结点的指针。删除链表头结点后面那个结点应该依次执行的语句是\_\_\_\_\_。
2. 为了增加空间的利用率和减少溢出的可能性, 在两个堆栈共享一片连续空间时通常将两个堆栈的栈底位置分别设置在这片空间的两端, 当\_\_\_\_\_时才会产生上溢。
3. 如果一棵二叉树有 1024 个结点, 其中 465 个是叶结点, 那么, 该二叉树中度为 1 的结点的个数是\_\_\_\_\_。
4. 已知在一棵二叉树中, a 是 b 的祖先结点, 若要通过遍历操作找到从 a 到 b 的路径, 则在前序遍历、中序遍历、后序遍历和按层次遍历这 4 种遍历方法中, 应该选择\_\_\_\_\_。
5. 如果从无向图中的任意一个顶点出发进行 1 次深度优先搜索便可以访问到图中的所有顶点, 这样的图一定是\_\_\_\_\_。
6. 在顺序表(b, c, d, e, f, g, q, r, s, t)中采用折半查找法查找元素 b 的过程中, 被比较过的元素依次为\_\_\_\_\_。
7. 所谓 m 阶 B-树中的 m 是指\_\_\_\_\_。
8. 若 n 个关键字互为同义词, 并且采用线性探测再散列法处理冲突, 则将这组关键字散列到一个散列空间中, 需要进行的探测次数为\_\_\_\_\_。
9. 在插入排序法、选择排序法、泡排序法和快速排序法这 4 种排序方法中有一种方法, 若初始时最小值元素位于待排序序列的最后, 则在最后一趟排序开始之前, 所有元

素都不在其最终位置上, 这种排序方法是\_\_\_\_\_。

10. 若采用泡排序法对序列(tang, deng, an, wang, shi, bai, fang, liu)中元素按值从小到大进行排序, 则第 2 趟排序结束时的结果是\_\_\_\_\_。

## 二、简答(本题共 20 分, 每小题各 5 分)

1. 一个完整的算法通常应该具备哪 5 个最基本的特性? (可以不对各特性的具体意义进行解释)

2. 对于长度为  $n$  的线性表, 下面给出的 4 种操作中, 在顺序表上实现比在链表上实现效率更高的是哪一种? 请分别用大  $O$  符号表示的时间复杂度形式说明你的理由。

- (1) 输出线性表中第  $i$  个数据元素的值( $1 \leq i \leq n$ );
- (2) 交换线性表中第 1 个数据元素与第 2 个数据元素的值;
- (3) 依次输出线性表的  $n$  个数据元素;
- (4) 输出线性表中与给定值  $x$  相匹配的数据元素在线性表中的序号。

3. 具有  $n$  个结点且深度也为  $n$  的二叉树一共有多少种? 请具体说明你的结论。

4. 为什么在建立散列表时若采用线性探测再散列法处理散列冲突容易产生聚集(clustering)? 采用其他什么方法可以减少这种聚集?

## 三、问题求解(本题共 20 分, 每小题各 5 分)

1. 设有编号分别为 1, 2, 3 的三辆列车顺序进入一个栈式结构的车站站台, 请分别写出这三辆列车开出车站所有可能的顺序。

2. 已知对某二叉排序树进行前序遍历得到的前序遍历序列为 (60, 45, 35, 40, 50, 65, 75, 70), 请画出该二叉排序树。

3. 若某无向图一共有 16 条边, 并且有 3 个度为 4 的顶点, 4 个度为 3 的顶点, 其余顶点的度均小于 3, 则该无向图至少有多少个顶点? (请写出结论的求解过程)

4. 在设计快速排序法的非递归算法时, 通常利用了一个堆栈来记录待排序区间的首、尾两个端点的位置, 而实际上也可以利用其他数据结构(如队列)来代替这个堆栈。请说明其中的理由。

#### 四、算法设计（本题 15 分）

已知某具有  $n$  个顶点的有向图采用邻接表方法存储，其中，用以存储有向边信息的边结点类型为

```
typedef struct edge{
    int adjvex;          /* 存放某有向边的终止顶点在顶点结点中的位置 */
    struct edge *next;   /* 指向下一个边结点 */
}ELink;
```

用以存储顶点信息的顶点结点类型为

```
typedef struct ver{
    int degree;          /* 存放某顶点的度 */
    vertype vertex;      /* 存放一个顶点的数据信息 */
    ELink *link;         /* 指向以该顶点为出发点的第一个边结点 */
}VLink;
```

并且  $n$  个顶点结点构成一个数组  $G[0..n-1]$ 。请写一个算法，该算法依次求出图中各顶点的度，并分别存放在相应的顶点结点的 `degree` 域中。

#### 五、单项选择（本题共 20 分，每小题各 2 分）

1. 下列关于 C 语言的自增和自减运算符的使用中，正确的是\_\_\_\_\_。

- A.  $15++$ ;                      B.  $(x+y)--$ ;                      C.  $++(a-b)$ ;                      D.  $s++++t+++u++$ 。

2. 若有  $\text{int } x=10$ ;，则执行  $y=(2*8, x+=5)$  以后， $x$  和  $y$  的值分别是\_\_\_\_\_。

- A. 15 和 16;                      B. 15 和 15;                      C. 16 和 15;                      D. 16 和 16。

3. 对于下列 4 种输入函数，当用户要输入的字符串中含有空格字符时，应该使用\_\_\_\_\_。

- A. `scanf()`;                      B. `getchar()`;                      C. `gets()`;                      D. `getc()`。

4. 对于  $\text{char } s1[] = \{\text{"abcd"}\}$ ;  $s2[] = \{\text{'a', 'b', 'c', 'd'}\}$ ;，下列叙述中，正确的是\_\_\_\_\_。

- A. 数组  $s1$  与数组  $s2$  等价;  
B. 数组  $s1$  的长度与数组  $s2$  的长度相等;  
C. 数组  $s1$  的长度大于数组  $s2$  的长度;  
D. 数组  $s1$  的长度小于数组  $s2$  的长度。

5. 设已有如下定义和语句:

```
int a=4, b=3, *ptr, *qtr, *t;
ptr=&a; qtr=&b; t=qtr; qtr=NULL;
```

下列 4 条赋值语句中, 错误的是\_\_\_\_\_。

- A. \*qtr=0;;      B. t=ptr;;      C. \*ptr=a;;      D. \*ptr=\*t;。

6. 若从键盘上输入 abc def, 则下列程序的输出结果是\_\_\_\_\_。

```
#include <stdio.h>
#include <stdlib.h>
void main( )
{
    char *ptr, *qtr;
    ptr=(char *)malloc(sizeof(char)*20);
    qtr=ptr;
    scanf("%s %s", ptr, qtr);
    printf("%s%s", ptr, qtr);
}
```

- A. defabc;      B. defdef;      C. abcdef;      D. abcabc。

7. 已知变量定义如下:

```
struct{
    float x, y;
    char s[10];
}point, *ptr=&point;
```

下列 4 个表达式中, 错误的是\_\_\_\_\_。

- A. ptr->x=2.0;      B. (\*ptr).y=3.0;      C. point.x=2.0;      D. ptr->s="str"。

8. 若有宏定义“#define AREA(a, b) a\*b”, 则下列“宏调用”中, 正确的是\_\_\_\_\_。

- A. s=AREA(r\*r);      B. s=AREA(x\*y);  
C. s=c\*AREA((x=3.5), (y+4.1));      D. s=AREA。

9. 设下列程序经编译链接后生成的可执行文件是 test.exe。若运行该程序时输入带参数的命令行为 test abcd efg h3 k44<回车>, 则程序执行后的输出结果是\_\_\_\_\_。

```
#include <stdio.h>
#include <string.h>
void main(int argc, char *argv[ ])
```

```
{    int i, len=0;
    for(i=1; i<argc; i+=2)
        len+=strlen(argv[i]);
    printf("%d\n", len);
}
```

A. 6;                      B. 8;                      C. 12;                      D. 14。

10. 下列程序的功能是\_\_\_\_\_。

```
#include <stdio.h>
void main()
{    FILE *fp;
    fp=fopen("file.dat", "r+");
    while(!feof(fp))
        if(fgetc(fp)=='*'){
            fseek(fp, -1L, SEEK_CUR);
            fputc('#', fp);
            fseek(fp, ftell(fp), SEEK_SET);
        }
    fclose(fp);
}
```

- A. 将文件 file.dat 中的所有字符都换成#;
- B. 将文件 file.dat 中所有的\*都换成#;
- C. 查找文件 file.dat 中所有的\*;
- D. 查找文件 file.dat 中所有的#。

计算机/软件工程专业

每个学校的

考研真题/复试资料/考研经验

考研资讯/报录比/分数线

免费分享



微信 扫一扫

关注微信公众号

计算机与软件考研

## 六、简答（本题共 20 分，每小题各 4 分）

1. 在 C 语言中，`int a[10]; a++;` 是否正确？为什么？
2. 在 C 语言中，`strlen` 与 `sizeof` 的区别是什么？
3. 使用函数 `strcpy(char *s1, char *s2)` 应该注意哪些问题？
4. 在 C 语言中，什么是函数指针？什么是指针函数？
5. 递归程序在进行递归调用时通常用到递归工作栈，请问：通常情况下，该递归栈中需要保存哪些信息？

## 七、程序阅读填空（本题共 20 分，每小题各 4 分）

1. 若已有定义 `int a[10];`，则下列函数 `FUNC()` 的功能是在第一个循环中给数组的 10 个元素依次赋予 1, 2, 3, 4, 5, 6, 7, 8, 9, 10；在第二个循环中使数组 `a` 的后 5 个元素的值为前 5 个元素的逆序，即数组变成 1, 2, 3, 4, 5, 5, 4, 3, 2, 1。

为了使该函数正确、完整，请写出函数的空白处(方框内)应该填入的内容。

```
FUNC(int a[])
{
    int k;
    for(k=1; k<=10; k++)
        [ ] = k;
    for(k=0; k<5; k++)
        [ ] = a[k];
}
```

2. 下列函数 `int MINPRIME(int num)` 的功能是返回大于 `num` 的最小素数。例如：若 `num=13`，则函数返回素数 17；若 `num=29`，则函数返回素数 31。

为了使该函数正确、完整，请写出函数的空白处(方框内)应该填入的内容。

```
#include <stdio.h>
int MINPRIME(int num)
{
    int k, j;
    k=num+1;
    while(1){
        for(j=2; j<=k-1; j++)
            if([ ]) /* 如果 k 不是素数 */
                [ ];
        if(j>k-1)
            return k;
        else k++;
    }
}
```

3. 下列函数 `int INDEX(char str[], char substr[])` 的功能是返回子串 `substr` 在主串 `str` 中的位置；若主串 `str` 中不存在子串 `substr`，则函数返回 0。

提示：所谓一个子串在包含它的主串中的位置是指主串中首次出现的这个子串的第 1

个字符在主串中的位置。例如：若 str="Beijing and Nanjing", substr="jing", 则 substr 在 str 中的位置为 4。

为了使该函数正确、完整，请写出函数的空白处(方框内)应该填入的内容。

```
#include <stdio.h>

int INDEX(char str[ ], char substr[ ])
{
    int i, j, k;
    for(i=0; str[i]; i++)
        for(j=i, k=0; str[j]==substr[k]; j++, k++)
            if( ) /* 若主串中存在该子串 */
                return ( ); /* 返回子串在主串的位置 */
    return 0; /* 若主串中不存在该子串 */
}
```

4. 下列函数的功能是用字符串 str2 替换字符串 olds 中所有出现的子串 str1, olds 串中的其他字符不变，并且将形成的新字符串存放在 news 中。

例如：若 olds="123ab123x", str1="123", str2="45", 则 news="45ab45x"。

为了使该函数正确、完整，请写出函数的空白处(方框内)应该填入的内容。

```
void REPLACE(char *olds, char *str1, char *str2, char *news)
{
    char *p, *q;
    while(*olds!='\0'){
        for(p=olds, q=str1; *p!='\0' && *q!='\0' && ( ); p++, q++);
        /* 在 olds 中查找与 str1 相匹配的子串 */
        if(*q!='\0') /* 若在 olds 中未找到与 str1 相匹配的子串 */
            *news++=( );
        else{ /* 若在 olds 中找到与 str1 相匹配的子串 */
            for(q=str2; *q!='\0'; q++)
                *news++=*q;
            olds=p;
        }
    }
    ( );
}
```

5. 下列程序的功能是找出由数字字符组成的文本文件 file.dat 中的所有不同整数，并且分别将这些整数存放于一维数组 a 中。

约定：该文本文件中的各整数之间以空格字符或者 Tab(制表符)或者回车符分隔，并且假定文件中的不同整数不超过 1000 个。

为了使该程序正确、完整，请写出程序的空白处(方框内)应该填入的内容。

```
#include <stdio.h>
#define N 1000
void main()
{
    FILE *fp;
    int a[N], i, num, count;
    if((fp = ) == NULL){           /* 打开文本文件 */
        printf("Can not open file!\n");
        exit(0);
    }
    count=0;
    while(fscanf() == 1){       /* 从文件中读一个整数 */
        a[count]=num;
        for(i=0; a[i]!=num; i++);
        if()
            count++;
    }
    fclose(fp);
}
```

#### 八、程序设计（本题 15 分）

请编写一程序，该程序的功能是计算并输出某子串 substr 在主串 str 中出现的次数。

要求：

(1) 把求子串在主串中出现次数的过程编写为一个独立的函数：

```
int STRCOUNT(char *str, char *substr)
```

(2) 若主串 str 中未出现子串 substr，则函数 STRCOUNT(char \*str, char \*substr)返回 0。

(3) 在主函数中通过键盘输入方式分别给 str 与 substr 赋值，并且所有涉及到字符串的操作均通过指针完成。