

西南交通大学 2012 年全日制硕士研究生入学试题

试题名称：数据结构与程序设计

一、单项选择题

1. C 语言规定，在一个源程序中，main 函数的位置（ ）。
A. 必须在最开始 B. 必须在调用库函数的后面
C. 可以任意 D. 必须在最后
2. 若 x、i、j 和 k 都是 int 型变量，则计算下面表达式后，x 的值是（ ）。 $x=(i=4, j=16, k=32)$
A. 4 B. 16 C. 32 D. 52
3. 设变量 a 是整型，f 是实型，i 是双精度型，则表达式 $10 + 'a' + i * f$ 的值的数据类型为（ ）。
A. int B. float C. double D. 不确定
4. 已有定义 $\text{int } a=2;$ 和输出语句： $\text{printf}(\text{"\%81x"}, a)$ 以下正确的叙述是（ ）。
A. 整型变量的输出格式符只有 %d 一种
B. %x 是格式符的一种，它可以适用于任何一种类型的数据
C. %x 是格式符的一种，其变量的值按十六进制输出，但 %81x 是错误的
D. %81x 不是错误的格式符，其中数字 8 规定了输出字段的宽度
5. 下面关于运算符优先顺序的描述中正确的是（ ）。
A. 关系运算符 < 算术运算符 < 赋值运算符 < 逻辑运算符
B. 逻辑运算符 < 关系运算符 < 算术运算符 < 赋值运算符
C. 赋值运算符 < 逻辑运算符 < 关系运算符 < 算术运算符
D. 算术运算符 < 关系运算符 < 赋值运算符 < 逻辑运算符
6. 若运行以下程序时，从键盘输入数据 2473<CR>（<CR>表示回车），则下面程序运行结果是（ ）。

```
#include<stdio.h>
main()
{int c;
while((c=getchar())!='\n')
switch(c- '2')
{case 0:
case 1: putchar(c+4);
case 2: putchar(c+4); break
case 3: putchar(c+3);
default: putchar(c+2); break
}
printf( "\n" );
}
```


A. 668977 B. 668966 C. 66778777 D. 6688766
7. 执行语句 $\text{for}(i=1; i++<4;)$ 后变量 i 的值是（ ）。
A. 3 B. 4 C. 5 D. 不定
8. 下面程序（ ）。（每行程序前面的数字表示行数）。
1. main()
2. {

```
3. int a[3]={0};
4. int i;
5. for(i=0;i<3;i++) scanf( "%d" ,&a[i]);
6. for(i=1;i<4;i++) a[0]=a[0]+a[i];
7. printf( "%d\n" ,a[0]);
8. }
```

- A.,没有错误 B.第三行有错误
C.第五行有错误 D.第六行有错误

9.下面是对 s 的初始化，其中不确定的是（ ）

- A.char s[5]={ "abc" } B.char s[5]={ 'a' , 'b' , 'c' };
C.char s[5]=" "; D.char s[5]= "abcdef" ;

10.若用数组名作为函数调用的实参，传递给形参的是（ ）。

- A.数组的首地址 B.数组第一个元素的值
C.数组中全部元素的值 D.数组元素的个数

11.以下不正确的说法为（ ）

- A.在不同函数中可以使用相同名字的变量。
B.形式参数是局部变量
C.在函数内定义的变量只在本函数范围内有效
D.在函数内的复合语句中定义的变量在本函数范围内有效

12.以下程序有错，错误的原因是（ ）。

```
main()
{ int*p,I;
  char*q,ch;
  p=&I;
  q=&ch;
  *p=40;
  *p=*q;
  ...
}
```

- A.p 和 q 的类型不一致，不能执行*p=*q; 语句
B.*q 中存放的是地址值，一次不能执行*p=40
C.q 没有指向具体存储单元，所以*q 没有意义
D.q 虽然指向了巨日单元，但对应单元没有确定的值，所以不能执行*p=*q; 语句

13.已知函数的调用形式：fread(buffer,size,count,fp); 其中 buffer 代表的是（ ）。

- A.一个整型变量，代表要读入的数据项的总数
B.一个文件指针，指向要读的文件
C.一个指针，指向读入数据的存放地址
D.一个存储区，存放要读的数据项

14.数据结构的定义为 (D,S)，其中 D 是（ ）的集合。

- A.算法 B.数据操作 C.数据元素集合 D.逻辑结构

15.若某线性表最常用的操作是存取任一指定序号的元素和在最后进行插入和删除运算，则利用（ ）存储方式最节省时间。

- A.顺序表 B.双链表

C.带头结点的双循环链表 D.单循环链表

16.在一个具有 n 个结点的有序单链表中插入一个新结点并保持单链表仍然有序的时间复杂度是（ ）。

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

17.若已知一个栈的入栈序列是 $1, 2, 3, \dots, n$, 其输出序列为 $p_1, p_2, p_3, \dots, p_n$ 若 $p_1=n$, 则 p_i 为（ ）

A. i B. $n-i$ C. $n-i+1$ D.不确定

18.串是一种特殊的线性表，其特殊性表现在（ ）。

A.可以顺序存储 B.数据元素是一个字符
C.可以链式存储 D.数据元素可以是多个字符

19.稀疏矩阵一般的存储方式有（ ）两种

A.二维数组和三维数组
B.三元组和散列表
C.散列表和十字链表
D.三元组表和十字链表

20.树最适合用来表示（ ）

A.有序数据元素 B.无序数据元素
C.元素间具有分支层次关系的数据
D.元素间无联系的数据

21.按二叉树的定义，具有三个结点的二叉树有（ ）种。

A.3 B.4 C.5 D.6

22.设结点 x 和 y 是二叉树中的任意两个结点，若在先根序列中 x 在 y 之前，而在后根序列中 x 在 y 之后，则 x 和 y 的关系是（ ）。

A. x 是 y 的左兄弟 B. x 是 y 的右兄弟
C. x 是 y 的后代 D. x 是 y 的祖先

23. n 个顶点的连通图至少有（ ）条边

A.0 B. $n-1$ C. n D. $n+1$

24.下面的查找方式中，可以对无序表进行查找的是（ ）

A.顺序查找 B.二分查找
C.二叉排序树 D.B-树上的查找

25.若需在 $O(n\log_2 n)$ 的时间内完成对数组的排序，且要求排序是稳定的，则可选的排序方法是（ ）

A.快速排序 B.堆排序
C.直接插入排序 D.归并排序

二、填空题

1.若有定义：char c='\010'; 则变量 c 中包含的字符个数为____个。

2.若 x 为 int 型变量，则执行以下语句后 x 的值为_____。

$x=7;$

$x+=x-=x+x;$

3.int x,y,z;且 $x=3, y=-4, z=5$, 则表达式 $(x\&\&y) == (x||z)$ 的值为_____。

4.当运行一下程序时，从键盘上输入 right?<CR>(CR 表示回车)，则程序运行的结果是_____。

```
#include<stdio.h>

main()
{char c;
while((c=getchar())!= '?' )putchar(++c);
}
```

5.若有定义:inta[3][4]={ {1,2},{0},{4,6,8,10}}; 则初始化后,a[1][2]得到的初值是_____,a[2][1]得到的初值是_____。

6.在 C 语言中，一个函数一般由两部分组成，它们是_____和_____。

7.函数 swap (int x, int y) 可完成对 x 和 y 值的交换。在执行下面语句后。a[0]和 a[1]的值分别是_____, _____。

8.以下程序的运行结果是_____。

```
#define A 4
#define B(x) A*x/2

main()
{ float c,a=4.5;
c=B(a);
printf( "%5.1fn" ,c);
}
```

9.下面程序中有语句错误，错误的原因是_____

```
#include<stdio.h>

main()
{ int*p;
*p=100;
...
}
```

10.下面程序段的运行结果是_____

```
char s[80]="abcd";
char *sp=s;
sp++;
puts(strcat(sp,"ABCD"));
```

11.在 C 程序中，文件中的数据可以用_____和_____两种代码形式存放。

12.数据结构与数据类型的形式定义分别为：

Data-Structure=(D, R)

Data-Type=(D, R, P)

其中 D 代表_____, R 代表_____, P 代表_____

13.在一个单链表中删除 p 所指结点时，可以执行以下操作：

```
q=p->next;
p->data=p->next->data;
p->next=_____;
```

free(q)

14.循环队列顺序存储在数组 a 中（下标为 0..n-1，共有 n 个元素）、f 为当前队首元素所在数组下标，r 为当前队尾元素在循环队列意义下的下一个位置。假设队列中的元素个数总小于

n，求队列中元素个数的公式是：_____。

15.表达式求值是数据结构_____应用的一个典型例子。

16.已知二叉树中叶子结点数为 50，仅有一个孩子的结点数为 30，则总结点数为_____。

17.利用孩子兄弟表示法作为树的存储结构，可以将一颗树转换为_____。

18.一个图的存储实现中，_____表示法是最唯一的，_____表示法不是唯一的。

19.动态查找表和静态查找表的重要区别在于前者包含有_____和_____。运算，而后者不包含这两种运算。

20.设表的长度为 256，没块的长度为 8，设为等概率查找，则索引表采用顺序查找的平均查找长度为_____。

21.设待排序随机分布且元素个数充分多时，在插入排序、冒泡排序、快速排序、选择排序、归并排序中，平均比较次数最少的排序方法是_____，需要辅助存储空间最多的排序方法是_____。

三、简答题

1.下面程序的运行结果是_____。

```
#include<stdio.h>
void main(void)
{
    int m,n;
    m=30;n=-2;
    do{
        m++;n+=10;
        if(n%3==2)printf(“%5d%5d\n”,m,n);
    }while(m>n);
}
```

2.下面程序的输出是：_____。

```
#include<stdio.h>
main()
{int i,j,a[5][5];
  for(i=1;i<=4;i++)
    for(j=1;j<=4;j++) a[i][j]=(i/j)*(j/i);
  for(i=1;i<=4;i++)
    {for(j=1;j<=4;j++) printf(“%5d”,a[i][j]);
      printf(“\n”);
    }
}
```

3.下面的程序输出是_____。

```
#include<stdio.h>
void p(int a,int*b,int*c)
{a=4; *b=5; *c=6;
  printf(“%d %d %d\n”,a,*b,*c);
}
main()
{int a,b,c;
```

```
a=1;b=2;c=3;
p(a,&b,&c);
printf(“%d %d %d\n”,a,b,c);
}
```

4. 以下程序的运行结果是_____。

```
#include<stdio.h>
main()
{int i,k;
 for(i=0;i<4;i++)
 { k=sub(&i);
 printf(“%3d”,k);
 }
 printf(“\n”);
}
int sub(int*s)
{static int t=0;
 t=*s+t;
 return(t);
}
```

5. 输入一个字符，如果它是一个大写字母，则把它变成小写字母；如果它是一个小写字母，则把它变成大写字母，其它字符不变。请在【 】中填入正确的内容。

```
main()
{ char ch;
 scanf(“%c”,&ch);
 if(【 】)ch=ch+32;
 else if(ch>='a' &&ch<='z')【 】;
 printf(“%c\n”,ch);
}
```

①_____

②_____

6. 设有一个顺序栈 S，元素 s₁,s₂,s₃,s₄,s₅,s₆ 一次进栈，如果六个元素的出栈顺序为 s₂,s₃,s₄,s₆,s₅,s₁，则顺序栈的容量至少应为多少？

7. 对给定的一组权值 W={5,2,9,11,8,3,7}，试构造相应的哈夫曼树，并计算它的带权路径长度。

8. 已知一棵二叉树的中序遍历结果为：DBFEAGHCI，后序遍历结果为：DFEBHGICA。画出这棵二叉树，并写出它的前序遍历结果。

9. 按如下序列的给定顺序：20,25,30,15,18,5,11,8，画出 二叉排序树。如何一句二叉排序树得到该给定序列的一个有序序列。

四、完善算法和计算题

1. 编写一个程序，实现的功能是：首先从键盘的终端上输入 10 个数序存放于数组 a 中，然后输入一个数 x，并查找 x 是否在数组中，若在数组中，则输出在数组中的元素序号，否则给出未查找到的信息。要求查找过程用一函数来实现。

2. 输入一个四位正整数，然后按数字的相反次序。如：输入 9187，则输出 7819。又如：若

输入为 7000，则输出为：0007。要求：只能用一个整型变量接受键盘输入》

3.若二叉树结点指针类型定义如下：

```
typedef struct bt_node{  
    int data;  
    struct bt_node*left,*right;  
}BiTNode,*Bitree;
```

下面 C++函数用先根遍历算法复制一棵二叉树设原来的二叉树根为 t，复制的二叉树根为 bt，请填空。

BiTree Copy (BiTree t)//复制二叉树 t

```
{BiTree bt;  
    if(_____) bt=null;  
    else{  
        _____;  
        _____;  
        bt->lchild=Copy(t->lchild);  
        bt->rchild=Copy(t->rchild);  
    }  
    _____;  
}
```

//结束 Copy

4.假设称正读和反读都是相同的字符序列为“回文”，例如“abba”和“abcba”是回文，“abcde”和“ababab”则不是回文。试写一个算法判别读入的一个以 '@' 为结束符的字符序列是否为回文。