

2000年北京理工大学程序设计考研试题
 考研加油站收集整理 <http://www.kaoyan.com>

第一部分 程序设计 (共50分)

【1】 选择题 (15分,每题1.5分)

- 若变量已正确定义并赋值,符合C语言语法的表达式是[]
 A) a=a+7; B) a=7+b+c, a++ C) int(12.3%4) D) a=a+7=c+b
- 不合法的十六进制数是[]
 A) 0xFF B) 0Xabc C) 0x11 D) 0x19
- 若有运算符: >, *%, <<, %, sizeof, 则它们按优先级(由低至高)的正确排列次序为 []。
 A) *% -> << -> -> % -> sizeof B) << -> *% -> -> % -> sizeof
 C) *% -> -> << -> sizeof -> % D) *% -> -> << -> % -> sizeof
- 以下程序段的输出是[]
 A) |3.1415| B) | 3.0| C) | 3| D) | 3. |

```
float a=3.1415;
printf(" |%6.0f|\n", a);
```
- 以下程序的输出结果是[]
 A) 9 8 B) 8 9 C) 6 6 D) 以上三个都不对

```
#include "math.h"
main()
{ double a=-3.0, b=2;
  printf("%3.0f%3.0f\n", pow(b, fabs(a)), pow(fabs(a), b));
}
```

6. 以下程序的输出结果是[]
 A) 39 81 B) 42 84 C) 26 68 D) 28 70

```
main()
{
    int x, i;
    for (i=1; i<=100; i++)
    {
        x=i;
        if (++x%2==0)
            if (++x%3==0)
                if (++x%7==0)
                    printf("%d", x);
    }
    printf("\n");
}
```

7. 以下程序的输出结果是[]
 A) 11 B) 20 C) 21 D) 31

```
func(int a, int b)
{
    int c;
    c=a+b;
    return c;
}

main()
{
    int x=6, y=7, z=8, r;
    r=func((x--, y++, x+y), z--);
    printf("%d\n", r);
}
```

8. 以下程序的输出结果是[]
 A) 4 B) 6 C) 8 D) 10

```
main()
{
    int k=2, m=4, n=6;
    int *pk=&k, *pm=&m, *p;
    *(p=&n)=*pk*( *pm);
    printf("%d\n", n);
}
```

9. 以下程序的输出结果是[]
 A) 不确定的值 B) 3 C) 2 D) 1

```
main()
{ int n[2]={0}, i, j, k=2;
  for(i=0; i<k; i++)
    for(j=0; j<k; j++) n[j]=n[i]+1;
  printf("%d\n", n[k]);
}
```

10. 以下程序的输出结果是[]
 A) ABCDEFGHIJKL B) ABCD C) ABCDEFGHIJKLMNOP D) AETM

```
main()
{ char
 *alpha[6]={"ABCD", "EFGH", "IJKL", "MNOP", "QRST", "UVWX"};
 char **p;
 int i;
 p=alpha;
 for(i=0; i<4; i++) printf("%s", p[i]);
 printf("\n");
}
```

【2】 填空题 (12分, 每题3分)

1. 以下程序的输出结果是_____。

```
#include "stdio.h"
main()
{char b[]="ABCDEFG", *chp=&b[7];
 while (--chp>&b[0]) putchar(*chp);
 putchar('\n');
}
```

2. 以下程序段的输出结果是_____。

```
char s[20]="goodgood!"; *sp=s;
sp=sp+2;
sp="to";
puts(s);
```

3. 以下程序的输出结果是_____。

```

fun (int n,int *s)
{ int f1, f2;
  if (n==1||n==2) *s=1;
  else
  { fun(n-1, &f1);
    fun(n-2, &f2);
    *s=f1+f2;
  }
}

main( )
{ int x;
  fun(6, &x);
  printf("%d\n", x);
}
    
```

4. 以下程序的输出结果是_____。

```

void fun(int *s)
{ static int j=0;
  do
  { s[j]+=s[j+1];
    while(++j<2);
  }
}

main()
{ int k,a[10]={1,2,3,4,5};
  for(k=1; k<3; k++) fun(a);
  for(k=0; k<5; k++) printf("%d",a[k]);
}
    
```

【3】阅读程序题（12分，每空2分）

1. 下面程序调用 `getone` 函数开辟一个动态存储单元，调用 `assone` 函数把数据输入此动态存储单元，调用 `outone` 函数输出此动态存储单元中的数据。请填空。

```
#include "stdlib.h"
getone(int **s)
{ *s=(1: _____)malloc(sizeof(int)); }
    assone(int *s)
{ scanf("%d", 2: _____);}
outone int *b)
{ printf("%d\n", 3: _____);}
main()
{int *p;
  getone(&p);  assone(p); outone(p);
}
```

2. 下列程序输出矩阵: 1 2 3 4 5 请填空.

```
16 17 18 19 6
15 24 25 20 7
14 23 22 21 8
13 12 11 10 9
```

```
#include <stdio.h>
#define M 100
int a[M][M], n;
void print(n)
int n;
{int i, j, k;
  printf("The matrix looks like following:\n");
  for (i=0; i<n; i++)
  { for (k=0; k<4*n; k++)
    printf("-");
    printf("\n");
    for (j=0; j<n; j++)
    { if (j==0) printf("|");
      printf("%3d|", a[i][j]);
    }
    printf("\n");
  }
  for (k=0; k<4*n; k++)
```

```

        printf("-");
        printf("\n");
    }
    void main()
    { int i, j, count=0, loop;
      printf("Input n:\n");
      scanf("%d", &n);
      while(n>=0)
      { i=0; j=0; count=(4: _____);
        a[i][j]=count++;
        loop=0;
        while(loop<n/2)
        { while(++j<n-loop) a[i][j]=count++; j--;
          while(++i<n-loop) a[i][j]=count++; i--;
          while(--j(5: _____)) a[i][j]=count++; j++;
          while(--i>loop) a[i][j]=count++; i++;
          loop(7: _____);
        }
        if(n%2) a[n/2][n/2]=count;
        print(n);
        printf("Input a another n(n<0 to quit)?\n");
        scanf("%d", &n);
      }
    }

```

【4】程序理解题（5分）

1. ◆单独考试答题◆

阅读下列程序，写出运行结果。（5分）

```

typedef union
{ long i;
  int k[5];
  char c;

```

} DATE;

共 13 页 第 6 页

```

struct date
{ int cat;
  DATE cow;
  double dog;
} too;
DATE max;
main()
{ printf("%d\n", sizeof(struct date)+sizeof(max)); }
    
```

2. ◆全国统考答题◆

以下程序编译、连接后生成可执行文件 CPY.EXE。假定磁盘当前目录下有三个文本文件，其文件名和内容分别为：

文件名	内容
a	aaaa#
b	bbbb#
c	cccc#

在 DOS 当前目录下键入：copy a b<CR>（此处<CR>代表 ENTER 键），则程序输出是什么？

```

#include "stdio.h"
void fc(FILE *);
main(int argc, char *argv[])
{FILE *fp; int i=1;
 while(--argc>0)
 {fp=fopen(argv [i++], "r");
  fc(fp);
  fclose(fp);
 }
void fc(FILE *ifp):
{char c;
while((c=getc(ifp)!='#') putchar(c-32);
}
    
```

【5】编程题（6分）**1. ◆单独考试答题◆**

设有以下结构类型说明：

```
struct stud
{
    char num[5], name[10];
    int s[4];
    double ave;
};
```

请编写函数 `readrec` 把 50 名学生的学号、姓名、四项成绩以及平均分放在一个结构体数组中，学生的学号、姓名和四项成绩由键盘输入，然后计算出平均分放在结构体对应的域中。

2. ◆全国统考答题◆

编写程序 `outch`，程序读入一行字符，根据命令行中的参数不同的输出。
若有命令行：`outch -2` 程序对所读入的一行字符输出最后两个字符。
若有命令行：`outch +6` 程序对所读入的一行字符输出开头 6 个字符。
若命令行中没有参数，则隐含规定输出最后 10 个字符。
为简单起见，命令行的参数中只含一个数字。

第二部分 数据结构部分 (50分)

一. 单项选择题 (10分)

1. 若某表最常用的操作是在最后一个结点之后插入一个结点或删除最后一个结点, 则采用_____存储方式最节省运算时间。

- (1) 单链表 (2) 双链表
(3) 单循环链表 (4) 带头结点的双循环链表

2. 一个栈的输入序列为 1 2 3 4 5, 则下列序列中不可能的输出序列是_____。

- (1) 2 3 4 1 5 (2) 5 4 2 3 1
(3) 2 3 1 4 5 (4) 1 5 4 3 2

3. 数组 $A[1..5, 1..6]$ 的每个元素占 5 个单元, 将其按行优先的次序存储在起始地址为 1000 的连续内存单元中, 则元素 $A[4][5]$ 的地址为_____。

- (1) 1140 (2) 1145
(3) 1120 (4) 1125

4. 对二叉树的结点从 1 开始进行连续编号, 要求每个结点的编号大于其左右孩子的编号, 同一结点的左右孩子中, 其左孩子的编

号小于其右孩子的编号，则可采用_____次序的遍历实现编号。

- (1) 先序
- (2) 中序
- (3) 后序
- (4) 从根开始按层遍历

5. 数据表中有 10000 个元素，如果仅要求出其中最大的 10 个元素，则采用_____算法最节省时间。

- (1) 堆排序
- (2) 希尔排序
- (3) 快速排序
- (4) 直接选择排序

二. 填空题 (10 分)

1. 带头结点的双循环链表 L 为空表的条件是_____。

2. 队列的特点是_____。

3. 对矩阵压缩存储是为了_____。

4. 对有 12 个元素的有序表 A[1..12] 作二分查找，则为查找元素值等于 A[4] 的元素，所比较过的元素下标依次为_____。

5. 若用数组 S[1..100] 作为两个栈 S1 和 S2 的共用存储结构，对任何一个栈，只有当 S[1..100] 全满时才不能作入栈操作。为这两

个栈分配空间的最佳方案是：S1 的栈底位置设为_____，S2 的栈底位置设为_____。

三. 请解答下列问题 (18分)

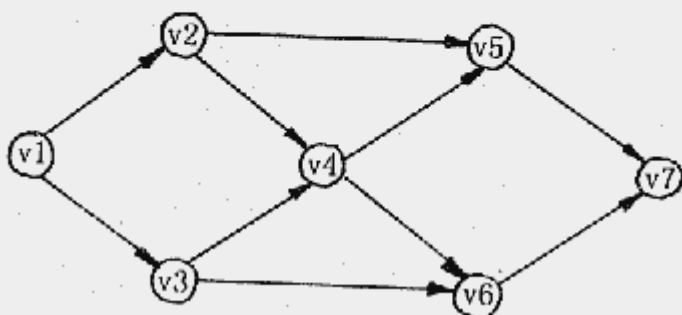
1. 试举一例，说明对相同的逻辑结构，同一种运算在不同的存储方式下实现，其运算效率不同。

2. 对于如图 1 所示的图

(1) 画出该图的邻接表。

(2) 给出该图的一个拓扑序列。

(3) 在你所给出的邻接表存储结构下执行拓扑排序算法，画出算法执行过程中，结点入度域的变化状态。



3. 对如图 2 所示的 3 阶 B 树依次执行下列操作，画出每步操作的结果。

- (1) 插入 70 (2) 插入 30 (3) 删除 150

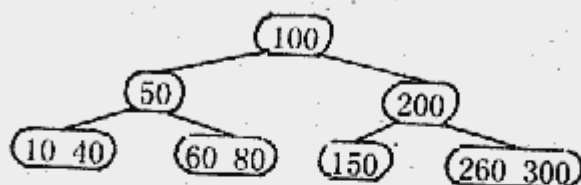
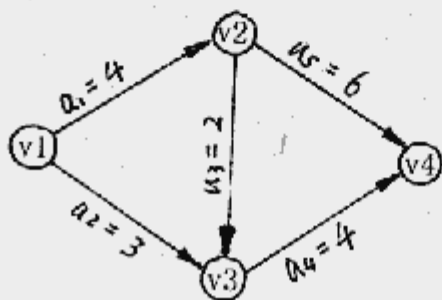


图 2

4. 对于如图 3 所示的 AOE 网，求出各活动的最早开始时间和最晚开始时间，并列出的所有的关键活动。



四、算法设计（12分）

1. 已知数组 $A[1..n]$ 的元素类型为整型，设计算法调整 A ，使其左边的所有元素小于零，右边的所有元素大于等于零。（要求算法的时间复杂度和空间复杂度均为 $O(n)$ ）

2. 设计算法将一个带头结点的单链表 A 分解为两个具有相同结构的链表 B 、 C ，其中 B 表的结点为 A 表中值小于零的结点，而 C 表的结点为 A 表中值大于等于零的结点（链表 A 的元素类型为整型，要求 B 、 C 表利用 A 表的结点）。

3. 设二叉树用二叉链表表示，设计算法求二叉树的高度。