

## 上海师范大学 2003 年硕士研究生入学考试试题

专业 教育技术学 计算机软件与理论 自然地理学

考试科目 程序设计 (424)

注意：请将答案写在答题纸上，按题意要求做，凡字迹潦草看不清的，一律不计分。

## 一、选择题 (40 分, 每题 2 分)

1. 在 PC 机中, 用 `sizeof( )` 检查 '`\n`' 在内存中占用的字节数为 ( )。  
① 1                      ② 2                      ③ 4                      ④ 不确定
2. Char 型常量在内存中存放的是 ( )  
① ASCII 代码值          ② BCD 代码值          ③ 内码值              ④ 十进制代码值
3. 设 `m, n, a, b, c, d` 均为 0, 执行 `(m = a == b) || (n = c == d);` 后 `m, n` 的值是 ( )。  
① 0, 0                      ② 0, 1                      ③ 1, 0                      ④ 1, 1
4. 设 `a=1, b=2, c=3, d=4`, 则表达式: `a++ < b ? a : c < d ? --a : --d` 的结果为 ( )。  
① 4                      ② 3                      ③ 2                      ④ 1
5. 设 `int x=5, y;` 执行语句 "`y = --x +--x/2`" 后, `y, x` 的值依次为 ( )。  
① 4, 3                      ② 4.5, 3                      ③ 5, 3                      ④ 6, 3
6. 若已定义 `x` 和 `y` 为 `double` 类型, 则表达式: `x=1, y=x+3/2` 的值为 ( )。  
① 1                      ② 2                      ③ 2.0                      ④ 2.5
7. 假定 `a` 和 `b` 为 `int` 型变量, 则执行以下语句后 `b` 的值为 ( )。  

```
a=1; b=10;
do { b-=a; a++; } while ( b-- < 0);
```

  
① 9                      ② -4                      ③ -2                      ④ 8
8. 设 `int a=8, b=7, c=6, x=1;` 执行下述语句后的值是 ( )。  

```
if (a>6) if (b>7) if (c>8) x=2; else x=3;
```

  
① 0                      ② 1                      ③ 2                      ④ 3
9. 若 `x` 为 `int` 型变量, 则执行以下语句后 `a` 的输出为 ( )。  

```
x=0xDEF;
printf ("%4d\n", x); printf ("%4o\n", x); printf ("%4x\n", x);
```

  
① 3567                      ② 3567                      ③ 3567                      ④ -3567  
6757                      6757                      6757                      -6757  
def                      def                      0xdef                      -def
10. 假定所有变量均已正确定义, 下列程序段运行后 `x` 的值是 ( )。  

```
a=b=c=0; x=35;
if (!a) x--; else if (b); if(c) x=3; else x=4;
```

  
① 34                      ② 35                      ③ 4                      ④ 3



002

11. 执行下列程序后的输出结果是 ( )。

```
#define M(a,b) (a) > (b) ? (a) : (b)
```

```
main ( )
```

```
{ int i=10, j=15; printf ("%d\n", 10*M(i, j)); }
```

- ① 150                      ② 100                      ③ 15                      ④ 10

12. 以下程序段给数组 a 的所有元素输入数据, 正确的填入答案为 ( )。

```
#include <stdio.h>
```

```
main ( )
```

```
{ int a[10], i=0;
```

```
while ( i<10) scanf("%d", — );
```

- ① a+i                      ② a+(i++)                      ③ &a[i+1]                      ④ &a[++i]

13. 假定 x 和 y 为 int 型变量, 则执行以下语句后 y 的值为 ( )。

```
for (y=1, x=1; y<=50; y++)
```

```
{ if (x>=10) break;
```

```
if (x%2 ==1) { x+=5; continue; }
```

```
x -= 3;
```

```
}
```

- ① 2                      ② 4                      ③ 6                      ④ 8

14. 设 a, b, c 都是 int 变量, 且 a=3, b=4, c=5, 下面表达式中值为 0 的是 ( )。

- ① 'a' && 'b'                      ② a <= b  
③ a||b+c&&b-c                      ④ !((a<b)&&!c||1)

15. 设有 int i = 010; 则 printf("%d,%d\n", ++i, i++); 的输出是 ( )。

- ① 011, 011                      ② 9, 9                      ③ 10, 8                      ④ 11, 11

16. 下面程序的运行结果输出为 ( )。

```
main( )
```

```
{ char str[10]="abc\0612def";
```

```
printf("%s", str);
```

```
}
```

- ① abc                      ② abc12def                      ③ abc 612def                      ④ 以上结果都不是

17. 语句 printf("%d\n", strlen("ats\n\x124\0xb\\")); 的输出是 ( )。

- ① 11                      ② 10                      ③ 9                      ④ 8

18. 若 i, j 已定义为 int 型变量, 则以下程序段中内循环体的执行次数是 ( )。

```
for (i=5; i; i--)
```

```
for (j=0; j<4; j++) { ..... }
```

- ① 20                      ② 24                      ③ 25                      ④ 30

19. 若定义 int a[5], \*p=a; 对 a 数组元素内容的正确引用是 ( )。

- ① &a[5]                      ② a+2                      ③ \*(p+5)                      ④ \*(a+2)

20. 已知在 PC 机上, 定义一数组 a 如下, int a[3][4]={1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}; 若数组 a 的首地址是 500, 则 a[1]+3 的值应是 ( )。

- ① 507                      ② 508                      ③ 514                      ④ 8





```
#include <stdio.h>
long fact( int i )
{ static long k=1L;
  k=k*i;
  return k; }
main( )
{ int n, i, t;
  scanf("%d",&n);
  for ( i=1; i<=n; t=fact(i), i++) ;
  printf("%d!= %ld\n", i, t);
}
```

2. 阅读程序，写出程序的输出结果。

```
#include <stdio.h>
void fun(int *x);
void main( )
{ int x, s=0;;
  for (x=0; x<5; x++)
    { fun(&x); s+=x; }
  printf("s=%d", s);
}

void fun(int *x)
{ static int y=0;
  *x%2 ? *x++:y++;
}
```

3. 阅读程序，写出程序的输出结果。

```
#include <stdio.h>
#include <string.h>
main( )
{ char a[80]="AB", b[80]="LMNP";
  int i=0;
  strcat(a, b);
  while (a[i++]!='\0') b[i]=a[i];
  puts(b);
}
```

4. 阅读程序，写出程序的输出结果。

```
#include <stdio.h>
void main( )
{ int i, j, s[3];
  for (i=0; i<3; i++) s[i]=0;
  for (i=0; i<3; i++)
```



## 程 序 设 计 (424)

```

    for (j=0;j<3;j++) s[j]=s[i]+1;
    for (i=0;i<3;i++) printf(" %d",s[i]);
    getch();
}

```

5. 阅读程序, 写出程序的输出结果。

```

#include <stdio.h>
test( int n)
{ printf("%4d",n);
  if ( n<=50 )
    { test( 2*n+1 ); printf("%4d",n); }
  return 0;
}
main( )
{ test(1); }

```

## 四、程序设计题 (60 分, 各个空格 4 分)

1. 阅读下列程序说明和 C 程序, 将应填入其中 ① 处的字句, 写在答卷的对应栏内。

[程序说明]

函数 maxword( ) 从给定的两个由英文单字组成的字符串 s 和 t 中, 找出其中都包含的最长的相同单字(同一字母的大小写视作不同的字符)。约定单字全由英文字母组成, 单字之间由一个或多个空白字符分隔。

程序采用以下算法思想: 自左至右顺序扫视字符串 s, 逐个找出单字(单字开始位置和单字长度), 当该单字的长度比已找到的单字更长时, 就从头至尾扫视字符串 t, 在从 t 中找出与该单字长度相等、字符相同的单字后, 登记该单字的开始位置和长度, 并回到 s, 在其中找下一个更长的单字, 上述寻找过程直至字符串 s 扫视接结束, 最后输出找到的单字。

[程序]

```

#include <stdio.h>
#include <string.h>
maxword(char *s, char *t)
{ char *res, *temp, chs, cht;
  int i, j, found, maxlen=0;
  while ( *s!='\0' )
  { while ( *s==' ' ) s++;
    for ( i=0; ① ; i++ );
    if ( i > maxlen )
    { chs = s[i]; ② ;
      temp=t; found=0;
      while (*temp!='\0' && !found)
      { while (*temp==' ' ) temp++;

```



006

```

        for (j = 0; ③; j++)
            if ( ① )
                cht=temp[j]; ④;
                if ( strcmp(s, temp) == 0 )
                { ⑤ = i;
                  res=s; found=1;
                }
                temp[j] = cht;
        }
        temp = &temp[j];
    }
    s[i] = &s[i];
}

if ( maxlen == 0 ) printf("there is no same word\n");
else { chr=res[maxlen]; res[maxlen]='\0';
      printf("%S\n", res); res[maxlen] = chr;
    }
}

char s[] = "This is C programming test", t[] = "This is a test for C programming";
main( )
{ maxmord(s, t); }

```

2. 阅读下列程序说明和 C 程序, 将应填入其中 ② 处的字句, 写在答卷的对应栏内。

[程序说明]

程序中的函数 qSort (NODE \*p) 是对给定的链表进行快速排序。函数在快速排序过程中, 以链表的首结点为基准, 把链表中的其它结点按小于、等于和大于基准点值分拆成为三个链表, 其表头分别为 lHead、mHead 和 hHead, 再递归地对其中小于和大于的两个链表分别进行快速排序。然后将这三个链表链接。

[程序]

```

#include <stdio.h>
#include <stdlib.h>
typedef struct node { int data;
                     struct node *link;
                     } NODE ; /* 链表结点类型 */

void write(NODE *p)
{ while (p) { printf( "%4d", p->data);
                p = p->link; }
  printf("\n\n");
}

NODE *blink(int *a, int n) /* 由数组生成链表 */
{ NODE *q, *h=NULL;

```



## 程 序 设 计 (424)

```

while (a != 0)
{
    ①;
    q->data = *a++;
    q->link = h;
    h = q;
}

return h;
}

NODE *qSort( NODE *p)    /* 快速排序函数 */
{
    NODE *lHead, *lTail, *mHead, *mTail, *hHead, *hTail, *q;
    if ( p==NULL) return  NULL;    /* 若是空链表结束 */
    lHead = lTail = hHead = hTail = NULL;    /* 小于和大于基准值的链表首末指针为空 */
    mHead = mTail = p;    /* 基准结点归入等于链表 */
    mTail->link = NULL;
    p = p->link;    /* 准备考察链表的下一结点 */
    while ( p!= NULL ) {
        q = p; p = p->link; q->link = NULL ;
        if ( q->data < mHead->data )    /* 当前结点值小于基准值 */
            if ( lHead == NULL ) lTail = lHead = q; else ②;
        else if ( p->data == mHead->data )    /* 当前结点值等于基准值 */
            mTail = mTail->link = q;
        else    /* 当前结点值大于基准值 */
            if ( hHead == NULL )
                hTail = hHead = q; else hTail = hTail->link = q;
    }
    if ( lHead != NULL ) /* 对非空的小于基准值链表排序, 将等于基准值链表接在它之后 */
    {
        q = p = ③;
        while ( q->link != NULL ) q = q->link ;
        ④;
    } else p = mHead ;
    if ( hHead != NULL ) /* 大于基准值链表非空, 排序后接在等于基准值链表之后 */
        ⑤;
    return p;
}

void main( )
{
    int a[ ]={ 8,6,3,15,10,5,4,2,7,9 };
    NODE *h;
    h = bLink( a, 10 ); write(h);
    h = qSort(h); write(h);
}

```

装

订

线



008

3. 阅读下列程序说明和 C 程序, 将应填入其中 ① 处的字句, 写在答卷的对应栏内。

[程序说明]

有  $n$  个班级参加  $ns$  个项目的比赛。下面的程序从文件 `tin` 读入  $n$  ( $\leq 30$ )、 $ns$  ( $\leq 10$ ) 和全部班级各项目的得分, 计算出各班的总分, 并按总分降序的次序将每个班级总分及各项目的得分输出到文件 `t.out`。

为了避免排序可能要交换 `score[i][k]` 和 `score[j][k]` ( $0 \leq k < ns$ ), 程序另引入数组 `order`, 改上述交换为 `order[i]` 和 `order[j]` 的交换。

[程序]

```
#include <stdio.h>
#define Number 30
#define Terms 10
#define INF "t.in"
#define OUTF "t.out"
int score[Number][Terms]
int total[Number], order[Number];
main()
{
    int i, j, n, ns, t, FILE *fpt;
    if ((fpt=fopen(INF, "r"))==NULL; {printf("Can' nopen file %s\n", INF); exit(1);}
    fscanf(fpt, "%d %d", &n, &ns);
    for (i=0; i<n; i++)
    {
        for (j=0; j<ns; j++)
            fscanf(fpt, "%d", &score[i][j]);
        for (t=j=0; j<ns; j++)
            t+= score[i][j];
        total[i]=t; ①
    }
    fclose(fpt);
    for (i=0; i<n-1; i++)
        for (j=②; j<n; j++)
            if (③)
                { t=order[i]; order[i]= order[j]; order[j]=t; }
    fpt=fopen(OUTF, "w");
    for (i=0; i<n; i++)
    {
        fprintf(fpt, "%4d %7d : ", i+1, total[④]);
        for (j=0; j<ns; j++)
            fprintf(fpt, " %3d " ⑤);
        fprintf(fpt, "\n");
    }
    fclose(fpt);
}
```