

浙江大学 2017 - 2018 学年春夏学期

《程序设计专题》课程期末考试试卷

课程号: 211G0260 , 开课学院: 计算机学院

考试试卷: ☒ A 卷、B 卷 (请在选定项上打 ☒)

考试形式: ☒ 闭、开卷 (请在选定项上打 ☒) , 允许带 / 入场

考试日期: 2018 年 07 月 05 日, 考试时间: 120 分钟

诚信考试, 沉着应考, 杜绝违纪.

考生姓名: _____ 学号: _____ 所属院系: _____

(注意: 答题内容必须写在答题卷上, 写在本试题卷上无效)

试题一、单选题 (每小题 3 分, 共 30 分)

1. 关于算法的时间空间复杂度, 以下说法正确的是_____.
A. 时间复杂度和空间复杂度都是问题规模 n 的函数
B. 计算量等于 $8n^2+n$ 的算法是 n^2-3n+1 的算法复杂度的 8 倍
C. 空间复杂度为 $O(n^3)$, 表明该算法的问题规模与 n^3 成正比
D. 算法的时间复杂度越大空间复杂度也越大

2. 下面程序的输出结果是_____.

```
int main()
{
    struct st {
        int n;
        struct st *next ;
    } a[3] = {5, &a[1], 7, &a[2], 9, &a[0]}, *p=a;
    int i;
    for( i=0; i<5; i++ ) p = p -> next;
    printf("%d\n",p -> n);
    return 0;
}
```

A. 0 B. 5 C. 7 D. 9

3. 以下叙述中正确的是_____.
A. 任一个 C 语言源文件 (.c 文件) 必须包含 main 函数
B. 一个 C 语言源文件 (.c 文件) 可以包含两个以上 main 函数

- C. 在一个可以正确执行的 C 语言程序中，一个函数的原型声明可以出现任意多次
D. C 语言头文件（.h 文件）和源文件（.c 文件）都要进行编译
4. 如下关于图形交互操作的说法中不正确的是_____.
- A. MovePen(x, y)把画笔移到（x,y）的位置，接下来绘图以这个位置开始
B. DrawLine(dx, dy)是绘制一条以（dx, dy）为终点的线段，绘制完成后画笔的位置是（dx,dy）
C. DrawArc(r,start,360)绘制半径为 r 的圆，绘制完成后不改变绘图的起点位置
D. DrawTextString("Hello")从当前位置开始输出文本“Hello”，输出完后当前位置改变。
5. 程序中应避免“魔鬼数值”，魔鬼数值指程序中直接出现的文字量，如 100、3.1415，应该用符号常量来替代魔鬼数值。使用符号常量，下面说法中错误的是_____.
- A. 修改会变得更可靠
B. 修改会变得更简单
C. 程序可读性提高
D. 程序中所有出现的数值都用符号常量替代，包括 0 和 1
6. 假设已定义如下所示的函数 Larger()和 Smaller()（可用于比较两数大小），以及整型数组 a:
- ```
int Larger(int a, int b);
int Smaller(int a, int b);
static int a[10]={6,3,5,7,4,2,9,8,0,1};
```
- 现欲定义函数 Bubble()，通过调用它，可对整型数组 a 进行排序，则\_\_\_\_\_是函数 Bubble()的正确原型声明.
- A. void Bubble(int a[], int size, int (\*cmd)(int,int) );  
B. void Bubble(int a[], int size, int \*cmd(int,int) );  
C. void Bubble(int \*a, int size, int cmd(int,int) );  
D. void Bubble(int \*a, int size, int \*cmd(int,int) );
7. 若有宏定义:
- ```
#define N 2
#define Y(n) ((N+1)*n)
```
- 则执行语句 int z; z = 2*(N+Y(5)); 后，z 的值是_____.
- A. 语句有错
B. 不确定
C. 34
D. 70
8. 将 newnode 指向的新节点（newnode->next 为 NULL）插入到 head 单向链表的链头，正确的是_____.
- A. newnode->next = head;
B. head = newnode;
C. if(head==NULL)
 head = newnode;
 else{
 head = newnode
 newnode->next = head->next;
 };
D. if(head==NULL)
 head = newnode;
 else{
 newnode->next = head;
 head = newnode;
 };

9. 根据下面函数定义，调用 `f(4)` 得到的结果是_____。
- ```
int f(int n)
{ if (n) return f(n - 1) + n; else return n; }
```
- A. 0                      B. 4                      C. 10                      D. 以上均不是
10. C 语言的编译系统对宏命令的处理是\_\_\_\_\_。
- A. 在程序运行时进行的  
B. 在程序连接时进行的  
C. 和 C 程序中的其它语句同时进行编译的  
D. 在对源程序中其它成份正式编译之前进行的

## 试题二、改错题（共 20 分）

1. 在一个 C 源程序文件中，若要定义一个只允许本源文件中所有函数使用的全局变量，则该变量定义时要加关键字 `extern` 来限定。（2 分）
2. 在最坏情况下，选择、插入、冒泡、归并和快速等排序算法的时间复杂度都一样，是  $O(N^2)$ 。（2 分）
3. 下面的递归函数欲求 `m` 和 `n` 的最大公约数，请指出其中的错误并修正。（2 分）

|   |                                    |
|---|------------------------------------|
| 1 | <code>int gcd(int m, int n)</code> |
| 2 | <code>{</code>                     |
| 3 | <code>if (n)</code>                |
| 4 | <code>return gcd(n,m/n);</code>    |
| 5 | <code>else</code>                  |
| 6 | <code>return m;</code>             |
| 7 | <code>}</code>                     |

4. 以下程序对输入的字符串字符进行简单处理后输出。请指出其中的错误并修正。（4 分）

|    |                                        |
|----|----------------------------------------|
| 1  | <code>#include &lt;stdio.h&gt;</code>  |
| 2  | <code>#include &lt;stdlib.h&gt;</code> |
| 3  | <code>#define LINELENGH 81</code>      |
| 4  |                                        |
| 5  | <code>int main( )</code>               |
| 6  | <code>{</code>                         |
| 7  | <code>char *p,*q;</code>               |
| 8  | <code>p =malloc(LINELENGH);</code>     |
| 9  | <code>gets(p);</code>                  |
| 10 | <code>while (*p) {</code>              |
| 11 | <code>*p=*p+1;</code>                  |
| 12 | <code>putchar(*p);</code>              |
| 13 | <code>p++;</code>                      |
| 14 | <code>}</code>                         |
| 15 | <code>free(p);</code>                  |

|    |           |
|----|-----------|
| 16 | return 0; |
| 17 | }         |

5. 假设工作目录下有文件 **a1.h** 和 **a1.c**. 请找出代码中的错误并修正。要求保留函数中的各条语句。（5 分）

|   |                             |   |                                  |
|---|-----------------------------|---|----------------------------------|
| 0 | /*文件 a1.h*/                 | 0 | /*文件 a1.c*/                      |
| 1 | #include <stdio.h>          | 1 | #include <stdio.h>               |
| 2 | #define a1.h                | 2 | #include "a1.h"                  |
| 3 | #ifndef _A1_H_              | 3 |                                  |
| 4 |                             | 4 | void Print(){                    |
| 5 | void Print();               | 5 | puts("Morning World!");          |
| 6 | int SayHello(char, double); | 6 | }                                |
| 7 |                             | 7 | int SayHello(char ch, double x){ |
| 8 | #ENDIF                      | 8 | printf("%c,%lf", ch, x);         |
|   |                             | 9 | }                                |

6. 指出下面程序的错误，不需要改正。程序输入一个字符串和一个字符，如果该字符在字符串中，就从该字符首次出现的位置开始输出字符串中的字符。例如，输入字符串 **This is the input string** 和字符 **t**，那么将输出 **the input string**。（2 分）

|    |                                              |
|----|----------------------------------------------|
| 1  | #include <stdio.h>                           |
| 2  | char *getone()                               |
| 3  | { char ch, str[80], *s = str;                |
| 4  | gets(str); /* 输入一个字符串 */                     |
| 5  | ch = getchar(); /* 读入一个字符 */                 |
| 6  | while ( *s )                                 |
| 7  | if ( *s == ch ) return s; /* 返回字符在字符串中的地址 */ |
| 8  | else s++;                                    |
| 9  | return (NULL);                               |
| 10 | }                                            |
| 11 |                                              |
| 12 | int main( )                                  |
| 13 | { char *p = NULL;                            |
| 14 | if( p = getone( ) ) /* 调用函数 getone( ) */     |
| 15 | printf("%s\n", p);                           |
| 16 | else                                         |
| 17 | printf("Not Found\n");                       |
| 18 | return 0;                                    |
|    | }                                            |

7. 以下程序对输入的 **n** 个学生的成绩信息，计算并输出每个学生的个人平均成绩。找出其中错误并修正。（3 分）

|    |                                                         |
|----|---------------------------------------------------------|
| 1  | #include <stdio.h>                                      |
| 2  | double count_average(struct student s)                  |
| 3  | {                                                       |
| 4  | return (s.math + s.english) / 2.0;                      |
| 5  | }                                                       |
| 6  | int main( )                                             |
| 7  | {                                                       |
| 8  | struct student{                                         |
| 9  | char name[10];                                          |
| 10 | int english, math;                                      |
| 11 | double average;                                         |
| 12 | };                                                      |
| 13 | int i, n;                                               |
| 14 | struct student s1;                                      |
| 15 | scanf("%d", &n);                                        |
| 16 | for(i = 1; i <= n; i++){                                |
| 17 | scanf("%s%d%d", &s1.name, &s1.math, &s1.english);       |
| 18 | s1.average = count_average(s1);                         |
| 19 | printf("name:%s, average:%.2f\n", s1.name, s1.average); |
| 20 | }                                                       |
| 21 | return 0;                                               |
| 22 | }                                                       |
| 23 |                                                         |

### 试题三、问答题（共 10 分）

1. 结构体变量作为函数参数的优缺点是什么？结构体指针作为函数参数与结构体变量相比优势是什么？
2. 假定下面代码中相关变量都已正确定义，且规定循环变量初始化、循环终止条件判断、循环变量自增、赋值均为基本操作，每种基本操作执行时间为常数 1。

```
for(column = 0; column < 10000; column++) {
 for(row = 0; row < 5; row++) {
 sum += table[row][column];
 }
}
```

接下来请完成下述任务：

- 1) 分析上面代码的执行时间；
- 2) 改写代码，使之优化；
- 3) 分析优化后代码的执行时间，与优化前代码比，是否能得出什么结论。

### 试题四、程序填空题（每空 2 分，共 26 分）

1. 如下是链表结点的结构类型定义：

```
typedef struct node {
 int data;
 struct node *next;
} NodeT;
```

请完成编写下列函数，它将不带头结点的单链表中的逆序结点删除，形成从小到大的有序序列。例如：若当前链表 h 为 3->3->2->6->1->4->7，将删除逆序结点 2、1、4，形成链表 3->3->6->7。

```
NodeT *order(NodeT *h)
{
 NodeT *p, *t;
 p = h;
 while (_____ 1 _____) {
 t = p->next;
 if (!t) break;
 else
 if (_____ 2 _____)
 p = p->next;
 else {
 p->next = t->next;
 _____ 3 _____ ;
 }
 }
 return h;
}
```

2. 以下程序用“比较计数”法对结构数组 a 按字段 num 进行升序排列。“比较计数”法的基本思想是：通过另一字段 con 记录 a 中小于字段 num 的元素的个数。待算法结束，a[i].con 就是 a[i].num 在 a 中的排序位置。请根据题意完成程序填空。

```
#include <stdio.h>
#define N 10
struct c{
 int num;
 int con;
} a[N];
int main()
{
 int i,j;
 for(i=0; i<N; i++) {
 scanf("%d",&a[i].num);
 _____ 4 _____ ;
 }
 for(i=N-1; i>=1; i--)
 for(_____ 5 _____; j>=0; j--)
 if(_____ 6 _____) a[j].con++;
 else a[i].con++;
}
```

```

 for(i=0;i<N;i++)
 printf("%d,%d\n",a[i].num,a[i].con);
 return 0;
}

```

3. 以下程序实现随笔画功能，即当鼠标左键按下拖动时，在窗口中随着鼠标位置画出轨迹，当鼠标左键抬起时则不画。请根据题意完成程序填空。

```

#include "graphics.h"
#include <winuser.h>
/*其他文件包含因篇幅所限未一一列出*/

void Painter(int x, int y, int button, _____ 7 _____)
{
 double cx = ScaleXInches(x);
 double cy = ScaleYInches(y);
 double dx,dy;
 static int isLeftButtonDown = 0;
 static double lx = 0.0, ly = 0.0;

 switch(event){
 case BUTTON_DOWN:
 if(button==LEFT_BUTTON) isLeftButtonDown = 1;
 lx = cx; ly = cy;
 break;
 case BUTTON_UP:
 if(button==LEFT_BUTTON) isLeftButtonDown = 0;
 break;
 case _____ 8 _____:
 if(isLeftButtonDown) {
 MovePen(lx,ly);
 dx = cx-lx;
 dy = cy-ly;
 DrawLine(dx,dy);
 lx = cx;
 ly = cy;
 }
 break;
 }
}

void Main(){
 _____ 9 _____;
 SetPenSize(1);
 SetPenColor("Black");
}

```

```

 _____10_____;
 }

```

4. 快速排序（简称快排）是一种分而治之的排序方法。首先任意选取一个元素，作为 **pivot**，然后设法将数组剖分为两部分：前半部元素  $\leq \text{pivot}$ ，后半部元素  $> \text{pivot}$ ；再对两个子数组进行递归排序，直到子数组元素个数不超过 1 个。阅读以下的快速排序程序，并将其填写完整。

```

/*a - 被排序的数组*/
/* low - 被排序的首元素下标*/
/* high - 被排序的末尾元素下标*/
void Qsort(int a[], int low, int high)
{
 int s = low, t = high, pivot;
 if (low >= high) return;
 pivot = a[(s+t)/2];
 a[(s+t)/2] = a[s];
 while (s < t) {
 while (t > s && a[t] >= pivot) _____11_____;
 a[s] = a[t];
 while (s < t && a[s] <= pivot) _____12_____;
 a[t] = a[s];
 }
 a[s] = pivot;
 Qsort(a, low, s - 1);
 _____13_____;
}

```

### 试题五、算法设计（共 14 分）

- 考虑用递归的方法对整型数组 **a** 前 **n** 个元素进行求和，请完成以下任务：
  - 写出递归算法的递归式和递归出口；
  - 写出实现该算法的函数定义（函数原型：int SumArr(int a[], int n); ）。
- 定义 **n** 个元素的集合  $S = \{ a_i, 0 \leq i \leq n-1 \}$  的“中位数”**m** 如下：如果把这 **n** 个数存入一个数组 **A[ n ]** 并从小到大排序，那么元素 **A[ (n-1)/2 ]** 就是中位数 **m**。现给定两个整数数组 **A** 和 **B**，它们都是 **n** ( $n > 0$ ) 个元素并且已经从小到大排序。现要找出 **A** 和 **B** 合并后的集合 **S** 的中位数 **m**。请完成：
  - 写出一个时间复杂性尽可能好的算法思想，并分析你的算法的时间复杂性；
  - 写出实现该算法的函数定义（函数原型：int Median(int A[ ], int B[ ], int n); ），要求加入足够注释。