

武汉大学计算机学院

2008-2009 学年度第 2 学期 2008 级

《高级语言程序设计》期末考试试卷 A 卷答案

姓名: _____ 学号: _____ 专业: _____

说明: 开卷考试, 答案请全部写在答题纸上, 写在试卷上无效。

未经主考教师同意, 考试试卷、答题纸、草稿纸均不得带离, 否则视为违规。

| 题号 | 一 | 二 | 三 | 四 | 五 | 总分 |
|----|----|----|----|----|----|-----|
| 总分 | 10 | 30 | 20 | 30 | 10 | 100 |

一. 单项选择题: (共 10 分, 每个小题 1 分)

- 以下说法错误的是 **B**。
A. 三种基本结构的共同点是“单入口单出口”。
B. goto 语句可以在程序中自由跳转。
C. 结构化程序设计三种基本结构的目的是增加程序的可读性和可维护性。
D. 结构化程序设计中划分模块的原则是模块“相对独立, 功能单一”。
- 在 C 语言中, 形参的缺省存储类是 **D**。
E. static F. register G. extern H. auto
- 设有如下宏定义:
#define test(t) t-4%t
则表达式 test(2)*test(5) 的值为 **B**。
I. -12 J. -2 K. 2 L. 0
- 执行语句 b=(a=2*10,a=a+4,(a=3*5,a*4),a+5); 后, a 和 b 的值分别是 **D**。
A. 24,29 B. 20,25 C. 15,25 D. 15,20
- 设有变量定义: int p=4; char k='a'; 则执行表达式((k+=2)>='c') || ((p*=3)<10) 后, 变量 k 和 p 的值分别为 **C**。
A. a 和 4 B. a 和 12 C. c 和 4 D. c 和 12
- 表达式 076432&~01777 的结果是 **D**。
A. 432 B. 0432 C. 76000 D. 076000
- 执行以下程序段后, 变量 m 的值为 **A**。
int a[8]={ {4,3,2,1},{7,6,5,4} };
int m,*p;
p = &a[0];
m = (*p + 1) + (p[2]) * ((p[7]));
A. 13 B. 11 C. 16 D. 19
- 设有变量定义: char a[30]; long int *p = (long int *) (a+3); 则以下与表达式 p+4 等价的是 **C**。
A. &a[7] B. a+16 C. a+19 D. a+28

9. 对于以下字符串, strlen(s)的值为: A
- (1) char s1[]="\t\"\\\\0abc\n";
 (2) char s2[]="\x69\33\n";
 (3) char s3[]="\t\082\n";
- A. 4、3 和 1 B. 4、3 和 3 C. 16、3 和 3 D. 16、3 和 10
10. 表达式 $2.5+7\%2-5/2*2$ 的结果为 C。
- A. 3.5 B. 1.5 C. -0.5 D. -1.5

二. 程序阅读与分析 (共 30 分, 每个小题 10 分)

11. 阅读以下程序, 请写出程序运行结果 (10 分)

/*源程序: A-11.C*/

```
#include <stdio.h>
#include <stdlib.h>
int main(void)
{
    int f0=2,f1=3,f2;
    int m=3;
    printf("%d\t%d\t",f0,f1);
    do{
        f2=f0+f1;
        printf("%d\t",f2);
        f0=f1;
        f1=f2;
        m++;
    }while(m<=10);
    return 0;
} /*end main*/
```

答案: 2 3 5 8 13 21 34 55 89 144

12. 请指出以下语句或语句段的错误, 并改正错误 (10 分, 每个小题 2 分)

a) double x[10]; int a; for(a=0; a<=10; a++) scanf("%lf", &x[a]);

答案: 数组越界, a<=10 改为 a<10

b) long s=1; int n=1;

while(n<=10)

s *= n; n++;

答案: 循环体缺少复合语句标记;

s *= n; n++;改为 { s *= n; n++;}

或者 s *= n, n++;

或者 s *= n++;

c) char *s; gets(s);

答案: 指针变量未初始化, 引起访问非法指针的严重错误。

改法 1: char s 改为 char s[10];

改法 2: gets(s); 前面增加语句 s=(char *)malloc(10);

改法 3: char *s,m[20]; s=m; gets(s);

说明: 字符串长度不限。

d) switch(c){

case 'a': printf("The number is a\n");

case 'A': printf("The number is A\n");

default: printf("The number is not a or A\n"); break;

}

答案: case 分支后缺少 break 语句。

改法: 两个 case 分支的最后增加 break 语句。

e) char str[10]; gets(str);

执行程序时从键盘输入: C language ✓

(说明: □表示空格键, ✓表示回车键)

答案: 字符串长度定义过短, 无法存储结束标记。

改法 1: 字符串 str 定义时长度至少应该为 11

改法 2: 键盘输入中去掉空格键

13. 现有递归函数 product 如下所示:

```
long int product(int a[ ], int n)
{
    int k;
    if(n==1) k = 0;
    else k = product(&a[1], n-1);
    return a[0] + k;
} /*end product*/
```

假设有定义: int prime[] = {2,3,5,7,11,13,17,19,23,29,31,37,41};

请问: 调用函数 product(prime, 5)后, 函数的返回结果是什么? 请分析并说明函数 product 的功能。

答案: 计算数组 a 的前 n 项之和, 结果为 28。

如果结果不正确, 但是程序执行流程分析正确, 可以适当给分 (2~ 6 分)

三. 子程序实现题 (每小题各 10 分, 共 20 分)

14. 设计一个函数 belongSet(s, c), 该函数用来判断元素 c 是否出现在集合 s 中; 如果 c 是集合 s 的元素, 返回 true(1); 否则, 返回 false(0)。

说明:

- 其中, s 为字符数组, 表示一个集合, 其全集为大小写字母 (52 个字母) 构成的集合。c 为字符型变量, 取值为大小写字母 (52 个字母) 之一。
- 例如: s 表示集合 {'a', 'x', 'D', 'Z'}, c 表示元素 'x'。则函数调用 belongSet(s,c); 返回值为整数 1。
- 请首先确定并简要说明字符数组 s 表示由 52 个字母元素构成集合的方法。

参考答案 1:

字符数组模拟位向量表示集合; 52 个元素, 下标 0~25 对应小写字母; 下标 26~51 对应大写字母;

int belongSet(char *s, char c)

```
{
    if(c>='a'&&c<='z')    return  s[c-'a'];
    else if(c>='A'&&c<='Z')    return  s[c-'A'+26];
    return 0;
}
```

参考答案 2:

字符数组模拟集合, 直接存储字母; 这里也有两种方法, 一是

52 个元素，下标 0~25 对应小写字母；下标 26~51 对应大写字母；
二是，顺序存储集合元素，空余区间存储'\0'；

```
int belongSet( char *s, char c)
{
    char x;
    if(c>='a'&&c<='z' ||c>='A'&&c<='Z')
        for(x=0;x<=51;x++)
            if(s[x]==c) return 1;
    return 0;
}
```

15. 设计一个函数 displayBase_7(number)，该函数用来将无符号十进制整数 number 转换为无符号七进制整数显示；

说明：

- number 为需要转换的十进制整数；转换后的七进制数据仅在显示器上显示，无需返回；例如：执行 displayBase_7(23); 语句之后，显示结果为 32。

参考答案：此程序有递归和非递归两种常见做法

```
void displayBase_7( unsigned int number)
{
    if(number<7) printf("%d", number);
    else
    { displayBase_7(number/7);
      printf("%d",number%7);
    }
}
```

四. 程序测试与分析（每小题各 10 分，共 30 分）

16. 以下语句段的功能是：打开文件 d:\student.dat

```
if(( fp = fopen("student.dat","rb"))==NULL)
{ printf("文件%s 打开错误!\n", filename);
  exit(1);
}
```

程序实际运行时，出现“文件 student.dat 打开错误”的执行结果。请分析并说明上述语句段出现打开文件错误的可能原因；并说明如何改正错误。

答案：文件不存在 或者 文件目录指定错误

改正：

- (1) 检查确保该文件存在，而且有读权限
- (2) fopen("student.dat","rb") 改为 fopen("d:\\student.dat","rb")

17. 假设 main 函数中已经创建了如图 1 所示的单链表：

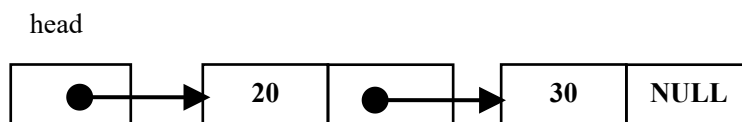


图1 函数调用前 链表 示意图

其中，链表定义如下所示：

```

struct node { int data; struct node *next; } head;
struct node 为链表结点的数据类型，head 为链表头指针。

```

函数 print_insert()的功能是：遍历链表 head，将每个节点数据值增加 12，然后在链表 head 头部插入一个取值为 n 的新结点；该函数返回值为插入结点后链表的结点个数。代码如下所示：

```

int print_insert( struct node *head, int n)
{
    int k=0;
    struct node *p=head,*new;
    while(p!=NULL)
    {
        p->data += 12; p=p->next; k++;
    }
    new=(struct node*)malloc(sizeof(struct node));
    if( new!=NULL)
    {
        new->data = n;
        new->next = head;
        head = new;
        k++;
    }
    return k;
}
/*end print_insert*/

```

执行以下函数调用语句

```
print_insert(head,11);
```

请问：

- 请用图示的方法说明执行上面的函数调用语句之后链表的结果。
- 执行上面的语句，分析是否真的可以在链表 head 中插入取值等于 11 的结点？如果不能插入取值为 11 新结点，请说明出错的原因，并改正错误。

答案：无法插入新结点，原因 1:head 定义错误，应该改为指针类型；原因 2：形参类型错误，无法带出链表头指针值到主调函数。

改正：错误 1：head 的定义 改为

```
struct node { int data; struct node *next; } * head;
```

错误 2

改法 1：形参 head 改为 struct node **phead，函数代码中所有 head 改为 *phead。

改法 2：去掉第 1 个形参定义，将头指针 head 定义为全局变量。

18. 根据以下公式计算 π 的值：

$$\pi = 4 - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \dots$$

程序 A-18.C 的功能是：显示用几项相加可以等于 3.1415 为止：

```
/*源程序：A-18.C*/
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
int main(void)
{
    double n=1,pi=4,sign=-1;
    int m=1;

    do{
        n= sign*(fabs(n)+2);
        pi += 4/n;
        m++;
        sign *=-1;
    }while(pi != 3.1415);
    printf("m = %d\n", m);
    return 0;
} /*end main*/
```

执行程序 A-18.C 时，程序出现无限循环，现象如下所示：黑屏、无任何显示、不能输入任何数据，键入 CTRL_break 可以强制终止程序执行状态。

请问：程序 A-18.C 出现哪种错误？请分析出错的原因，并改正错误。

答案：错误点（加粗位置），原因：C 语言中实数的可表示误差问题。

改正 **`fabs(pi-3.1415)>=0.00005`**

说明：误差设置至少应该小于 0.00005

五. 算法设计（任选一题完成，共 10 分）

说明：

- 19~21 题中仅需任意**选做一题**。完成多个题目，则以答卷纸上写在前面的一个题目为准评分。
- 本题目以描述算法思路为主；**不必完整写出函数程序的代码**，完整代码将不作为判分依据。
- 分析算法思路，说明算法中的重要数据类型的定义；
- 说明算法中的模块划分；各模块的功能、形式参数、函数返回值说明；并用伪代码、流程图或其它方法描述子程序的执行过程。

19. 编程完成队列操作：队列是以先进先出顺序访问的线性列表。队列的两个基本操作：入队，表示插入一个新数据到队列中，新数据放在队列尾部。出队，从队列首部取走一个数据。要求用数组模拟队列，编程实现入队和出队的操作，入队和出队数据都为整数。输入 0 表示执行出队操作，输入-1 表示程序执行结束，其余整数表示执行入队操作。

例如队列内容为“1、3、5”，则将数据 7 入队后，队列内容变为“1、3、5、7”。而执行一次出队操作之后，出队数据为 1，队列内容为“3、5、7”。

请注意：队列为空时，执行出队操作错误。队列为满时，执行入队操作错误。

具体要求如下：

- A. 请用图示的方式说明如何用数组表示一个队列;
- B. 给出用数组表示队列的 C 语言定义形式;
- C. 给出入队函数 `qstore()` 和出队函数 `qretrieve()` 的函数原型, 说明传入和传出信息的方式; 并简要描述这两个函数的算法。

参考答案:

需要定义 3 个标识符, 数组表示队列; 队头、队尾标记 (为整数类型或者指针类型)。

```
int q[SIZE], t1, t2;  
int qstore(int n) ;//n 入队操作, 返回 1 表示操作成功, 0 失败  
int qretrieve( ) ;//返回出队数据
```

20. 编写程序完成文件合并操作: 假设有两个数据文件 `price1.dat` 和 `price2.dat`, 其中分别存入了各 90 个长整型的数据。编程实现对文件 `price1.dat` 和 `price2.dat` 和合并, 要求按照数据从小到大的顺序合并, 结果存放到文件 `result.dat` 中。

限制条件: 编程中允许使用的内存空间限制在 100 个长整型数据范围之内。

具体要求如下:

- A. 请说明利用 100 个长整型数据组成的数据完成两个数据文件 `price1.dat` 和 `price2.dat` 的排序和合并的基本思路。
- B. 说明程序中划分多少模块? 并给出每个模块的原型;
- C. 简要描述每个模块的算法。

参考答案:

基本思路: `price1.dat` 先排序; `price2.dat` 再排序; 二者合并;

至少 2 个函数: 排序函数; 需要写明采用何种排序算法; `void sort(long int *s);`
文件合并函数; `void function(FILE *fp1, FILE *fp2, FILE *fp3);`