（装订线内不要答题）

2022 ～2023 学年第 1 学期

《　　程序设计　　》期末考试试卷

A 卷　　共 10 页

课程代码： COMP120006.01　　**考试形式：** □半开卷 □开卷 ■闭卷

开课院系： 微电子学院、信息学院、计算机学院　　2022 年 12 月

（本试卷答卷时间为 120 分钟，答案必须写在试卷上，做在草稿纸上无效）

专业＿＿＿＿＿＿＿＿＿＿＿＿学号＿＿＿＿＿＿＿＿姓名＿＿＿＿＿＿＿＿

题号	一	二	三	四	五	总分
得分						

一、选择题（30 分，每题 2 分）

1.　如果希望调用某一函数时，使该函数内某一变量拥有最近一次退出该函数时所拥有的值，同时不希望其他函数访问该变量，则该变量应定义为什么存储类型

A、auto

B、register

C、extern

D、static

2.　以下是正确的 C 语言标识符是＿＿＿＿＿＿＿＿。

A、-int　　B、in1_3　　C、A_B!D　　D、const

3.　C 语言中，“\\xfds”在内存中占用的字节数是＿＿＿＿＿＿。

A、3　　B、4

C、5　　D、6

4.　下列运算符中优先级最高的是＿＿＿＿＿。

A、<　　B、+　　C、%　　D、 ++

5.　执行下列语句后输出的结果是＿＿＿＿＿。

int a[2][3] = { 1,2,3,4,5,6 };

```
int* p;
p= a[1];
printf("%d", p[2]);
```

A、4

B、5

C、6

D、超出数组

6. 下面程序中，对 pp 出生年月输出正确的是_____。

```
typedef struct date
{
    int year;
    int month;
    int day;
}* DATE;

typedef struct student
{
    long studentID;
    char studentName[10];
    char studentSex;
    DATE birthday;
    int score[4];
}STUDENT;
STUDENT pp;
```

A、printf(“%d,%d”, pp. DATE ->year,pp. DATE ->month)

B、printf(“%d,%d”, pp.birthday ->year,pp. birthday ->month)

C、printf(“%d,%d”, pp.birthday.year,pp. birthday.month)

D、printf(“%d,%d”,pp-> DATE.year,pp-> DATE.month)

7. 设 a 为整型变量，与表达式 10<a<15 相等的表达式是_____。

A、 1　　B、a==11||a==12||a==13||a==14

C、 0　　D、!(a<10) && !(a>=15)

8. 下面不能正确定义二维数组的选项是_____ 。

A、int a[2][]={{1,2},{3,4}};

B、int a[][2]={1,2,3,4,5};

C、int a[2][2]={{1},{2}};

D、int a[2][2]={1,2,3};

9. 若有以下语句，各选项中数值为 4 的表达式是_____。

```
int a[12]={1,2,3,4,5,6,7,8,9,10,11,12};
char c = ‘a’, d, g;
```

A、a[g-c]　　B、a[4]

C、a[‘d’-‘c’]　　D、a[‘d’-c]

10. 下面叙述中，错误的是_____。

A、对于实型数组，不可以在 scanf 与 printf 函数中直接用数组名对数组进行整体的输入或输出。

B、对于字符型数组，可以在 scanf 与 printf 函数中用%s 对数组的数组名进行整体的输入或输出。

C、对于字符型数组，可以用来存放字符串。

D、对于含有数组的结构体变量，可以在赋值语句中运用“=”进行整体的赋值。

11. 若有以下说明和语句，int c[4][5],(*p)[5];p=c;则正确访问 c 数组元素的表达式是_____。

A、p+1

B、*(p+3)

C、*(p+1)+3

D、*(p[0]+2)

12. 表示数组 int a[2][3]的第 i 行第 j 列元素地址的正确语句是_____。

A、*(a[i]+j);　　B、(a+i);

C、*(a+j);　　D、a[i]+j ;

13. 若有 int *p, a=4, n;下面正确的程序段是_____。

A、p=&n; scanf("%d", &p);

B、p=&n; scanf("%d", *p);

C、scanf("%d", &n); *p=n;

D、p=&n; *p=a;

14. 执行下列程序，输出结果是_____：

```
#include <stdio.h>
int x=5;
int func(int x1,int x2){
    extern int x;
    x1>x2?(x=3):(x=4);
    return x+x1;
}
void main(){
    printf("%d",func(6,7));
}
```

A、9

B、10

C、11

D、12

15. 要打开一个已经存在的用于修改的非空二进制文件“hello.txt”，正确的语句是 _____。

A、fp=fopen(“hello.txt”,”r”)

B、fp=fopen(“hello.txt”,”ab+”)

C、fp=fopen(“hello.txt”,”w”)

D、fp=fopen(“hello.txt”,”r+”)

二、程序阅读题（15 分，每题 3 分）

1. 下面程序的输出是。

```
#include   <stdio.h>
#define M 1+2
void main()
{
   char str[6]={‘a’,’b’,’\0’,’c’,’d’,’\0’};
int x = 8, *p = &x;
int b[5], i;
printf(“%s\n”,str);
for(i=0;i<=4;i++)    b[i]=i*3;
printf(“%d\n”,b[4]);
printf(“%d\n”,0<=x<=3);
printf(“%d\n”,2*M*3);
printf(“%d\n”,*p);
printf(“%c\n”,str[‘\0’]);
}
```

2. 请写出下列函数的输出。

```
#include <stdio.h>
#include<stdlib.h>
#include<string.h>
int main()
{

        int b[3][3];
        int c[3][3] = {0};
        int i, j, k;
        int a[3][3] = {{1,2,3},{4,5,6},{7,8,9} };
for (i = 0; i < 3; i++)
        for (j = 0; j < 3; j++)
        {
                b[j][i] = a[i][j];
        }
for (i = 0; i < 3; i++)
        for (j = 0; j < 3; j++)
        {
                for ( k = 0; k < 3; k++)
```

在C语言中，`fopen` 函数用于打开文件，其第二个参数指定了文件的打开模式。对于需要修改已存在的非空二进制文件，正确的模式应该是能够读写文件，并且不会导致文件内容被截断。

让我们分析每个选项：

A. `fp=fopen("hello.txt","r")`
这个选项以只读模式打开文件。在这种模式下，不能对文件进行写操作，所以这不是修改文件的正确方式。

B. `fp=fopen("hello.txt","ab+")`
这个选项以二进制模式打开文件，允许读写（`a` 表示追加，`b` 表示二进制，`+` 表示读写）。但是，`ab+` 模式通常用于在文件末尾追加数据，而不是修改文件的现有内容。

C. `fp=fopen("hello.txt","w")`
这个选项以写模式打开文件，这会截断文件，即删除文件中的所有现有内容。这不是修改文件的正确方式，因为它会导致数据丢失。

D. `fp=fopen("hello.txt","r+")`
这个选项以读写模式打开文件，允许对文件进行读写操作，且不会截断文件。这是修改已存在文件的正确方式。

因此，正确的语句是 D，`fp=fopen("hello.txt","r+")`。这个选项允许你读取和修改文件中的现有数据，而不会丢失任何内容。

```
                {
                    c[i][j] += a[i][k] * b[k][j];
                }
            }
        for (i = 0; i < 3; i++)
            for (j = 0; j < 3; j++)
            {
                printf("%d ", c[i][j]);
            }
    }
```

3. 写出下面程序执行后的输出结果。

```
void swap(int *a, int *b)
{ int *t;
t=a; a=b; b=t;
}
int main()
{ int i=2,j=5,*p=&i,*q=&j;
swap(p,q); printf("%d %d\n",*p,*q);
}
```

然而，这段代码中 `swap` 函数的实现有一个错误。在 `swap` 函数中，`t` 被声明为一个指针，但是在交换 `a` 和 `b` 之前，没有为 `t` 分配存储空间。这将导致未定义行为，因为 `t` 被用来存储 `a` 的值，而 `a` 和 `b` 都是指针。

4. 下面函数的功能是

```
typedef struct sListNode {
    int data;
    struct sListNode* next;
} SListNode;
SListNode* function1(SListNode* head)
{
    SListNode* prev = NULL;
    SListNode* current = head;
    while (current) {
        SListNode* next = current->next;
        current->next = prev;
        prev = current;
        current = next;
    }
    return prev;
}
```

5. 下面程序的输出结果是：

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

```
void fun(char* w, int n) {
        char t1,t2, * s1, * s2;
        s1 = w; s2 = w + n - 1;
        while (s1<s2) {
                t1 = *s1++;
                *s1 = *s2--;
                *s2 = t1;
        }
}
int main() {
        char p[8] = "1234567";
        fun(p, strlen(p));
        printf("%s",p);
        return 0;
}
```

三、代码改错题（以下程序段中有若干错误，请在不增删语句的情况下指明程序中的第几行有错误，并且写出正确的语句，每个错误 2 分，共 22 分）

1. 以下程序的运行结果为"u=4, v=8"。

```
/* 第 1 行 */ void func(int *a, int *b)
/* 第 2 行 */ {
/* 第 3 行 */           int *x;
/* 第 4 行 */           x = *b;
/* 第 5 行 */           *b = *a;
/* 第 6 行 */           *a = x;
/* 第 7 行 */ }
/* 第 8 行 */ int main()
/* 第 9 行 */ {
/* 第 10 行 */              int u=8, v=4;
/* 第 11 行 */              func(u, v);
/* 第 12 行 */              printf("u=%d, v=%d\n", &u, &v);
/* 第 13 行 */              return 0;
/* 第 14 行 */          }
```

2. 需要以下程序的运行结果为"34512"。

```
/* 第 1 行 */ int main() {
/* 第 2 行 */           int a[5] = { 1,2,3,4,5 };
/* 第 3 行 */           int i,index;
/* 第 4 行 */           for (i = 0;i <= 5;i++) {
/* 第 5 行 */               index = (3 + i) % 5;
```

```
/*  第 6 行  */              printf("%d", (a + index));
/*  第 7 行  */          }
/*  第 8 行  */          return 0;
/*  第 9 行  */ }
```

3. 完成递归函数，根据主函数中输入 digit 函数的 1234，得到打印输出 edcb。

```
/*  第 1 行  */ void digit(int n) {
/*  第 2 行  */          char ch;
/*  第 3 行  */          if (n < 10) {
/*  第 4 行  */              printf("%c", n + "a");
/*  第 5 行  */          }
/*  第 6 行  */          else {
/*  第 7 行  */              printf("%c", n % 10 + 'a');
/*  第 8 行  */              digit(n);
/*  第 9 行  */          }
/*  第 10 行  */ }
/*  第 11 行  */ int main() {
/*  第 12 行  */              digit(1234);
/*  第 13 行  */              return 0;
/*  第 14 行  */ }
```

4. 完成链表的删除节点操作。

```
/*  第 1 行  */ struct node {
/*  第 2 行  */          int data;
/*  第 3 行  */          struct node next;
/*  第 4 行  */ };
/*  第 5 行  */ typedef struct node* ptr;
/*  第 6 行  */ ptr delnode(ptr head, int n) {
/*  第 7 行  */          ptr tmp,p;
/*  第 8 行  */          if (head->data = n) {
/*  第 9 行  */              return head->next;
/*  第 10 行  */              }
/*  第 11 行  */              for (p = head;p->next != NULL;p = p->next) {
/*  第 12 行  */                  if (p->data == n) {
/*  第 13 行  */                      p->next = p->next->next;
```

```
/*  第 14 行  */                    return head;
/*  第 15 行  */                }
/*  第 16 行  */            }
/*  第 17 行  */        }
```

四、程序填空题（14 分，每空 2 分）

1. 列是用公式π＝4/1- 4/3 ＋ 4/5 －4/7 ＋……计算圆周率π的近似值。

```
#include <stdio.h>
#include <math.h>
#define EPS 1e-6
double PI(){
        double pi,t;
        int i,s;
        ____（1）____
        for(i=1,s=1;fabs(t)>EPS;i+=2,s=-1*s){/*fabs 函数返回绝对值*/
                ____（2）____
                pi+=t;
        }
        ____（3）____
}
int main(){
        printf("pi=%f\n",PI());
        return 0;
}
```

2. 下列程序的功能是把带辅助表元的链表中的词汇（word）和数量（number）信息输出到文件中，每个表元的信息占一行，每个单词长度不超过 100。

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
        （4）
struct Node{
        char word[MAXN];
        int number;
        struct Node *next;
};
void printDict(struct Node *h, char *fname){
        FILE * fp;
```

```
struct Node * p;
if ((fp=fopen(fname,"w"))==NULL){
                printf("Can't open %s\n",fname);
                ____（5）_____
}
for (____（6）____; p; p=p->next){
                fprintf(fp, "%s\t%d\n", p->word, p->number);
        /*函数 fprintf 为文件的格式化输出函数*/
}
```

（7）

}

五、　　算法编程题（共 19 分）

1. 根据已定义的主函数完成程序，实现如下功能的两个函数（6 分）：

1) 对输入的长度为 n 的数组，使用 sort 函数中的冒泡排序将其从大到小排序。

2) 使用 fac 函数，利用递归的方法求 1)得到的数组中的最大值的阶乘（不考虑溢出）

```
#include<stdio.h>
……
……
int main() {
int a[100], i, n;
scanf("%d", &n);
for (i = 0;i < n;i++) {
        scanf("%d", &a[i]);
}
sort(a,n);
printf("%d", fac(a[0]));
}
```

2. 利用已有的部分代码，完成实现以下功能的程序：假定有两个使用链表存储的多项式 A 和 B，求他们的和多项式 C 并打印。例如 $A = 3.2x^3 + 2.1x^1$, $B = 1.1x^4+1$，则 $C = 1.1x^4+3.2x^3 + 2.1x^1+1$（13 分）。注意：

（装订线内不要答题）

1) 多项式链表的表元定义为结构体 poly，包括：double 类型的系数、int 类型的指数，还有指向下一表元的指针。
2) 多项式链表 A，B 的表元顺序根据指数从大到小排列，要求和多项式 C 也满足该顺序要求。所有指数均为正整数。
3) 下面已给出部分主函数代码，输入函数 readpoly 视为已知，不需编写。直接实现结构体 poly、函数 sumpoly 与 outprint 即可。

```
#include<stdio.h>

#include<stdlib.h>

//请编写函数与结构体

int main() {
    struct poly * ha, *hb, *hc;//三个多项式的头
    ha = readpoly()//读入 ha，readpoly 视为已知，无需实现
    hb = readpoly()//读入 hb，readpoly 视为已知，无需实现
    //下列函数需要实现
    hc = sumpoly(ha, hb);
    outprint(hc);
    return 0;
}
```