

## 2020-2021 学年第一学期期中考试试卷

### 一、判断题

1-1 (A05-21)表示数学式  $3 < x < 5$  含义的 C 语言表达式可以是  $3 < x < 5$ 。(2 分)

◎T ◎F

1-2 (D01-19)执行语句 `printf("%x,%d,%c", '1', '1', '1');`，输出是 31,49,1。(2 分)

◎T ◎F

1-3 (A05-19)表达式 `!!6` 的值是 6。(2 分)

◎T ◎F

1-4 (D01-18)对于下列 for 循环语句，循环体一次也不执行。(2 分)

```
int i, k;
```

```
for(i=0,k=-1; k=1; i++,k++)
```

```
    printf("***");
```

◎T ◎F

1-5 (D02-4)在函数内部定义的变量，包括形参，称为局部变量。(2 分)

◎T ◎F

1-6 (B03-9)在 C 程序中，如果变量 x 和 y 已经正确定义，语句 `if(x=5) y++;` 没有编译错误。(2 分)

◎T ◎F

4-1 (HO6-16)下面程序的运行结果是:\_\_\_\_\_ (3 分)

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

```
int b=1;
```

```
void fun()
```

```
{
```

```
    int c=2;
```

```
    static int a=0;
```

```
    a=a+b+c;
```

```
    printf("%d#",a);
```

```
}
```

```
int main()
```

```
{
```

```
    int cc;
```

```
    for(cc=1;cc<4;cc++)
```

```
        fun();  
    }
```

4-2 (B00-16)以下程序运行后的输出结果是\_\_\_\_\_(3 分)

```
main ()  
{  
    int i=10,j=0;  
    do {  
        j=j+i; i--;  
    }while (i>5);  
    printf ("%d",j);  
}
```

4-3 (R00-07)若有以下程序执行后输出结果是\_\_\_\_\_。(3 分)

```
main ()  
{  
    int a=4, b=1, c=7,t=0;  
    if (a<b) t=a ; a=b ; b=t ;  
    if (a<c) t=a ; a=c ; c=t ;  
    printf ( "%d#%d#%d",a, b, c);  
}
```

4-4 (u10-6)下面程序运行后的输出结果是\_\_\_\_\_(3 分)

```
#include "stdio.h"  
  
int main()  
{  
    int a=0,b=0,c=0,i;  
    for(i=0;i<4;i++)  
        switch(i)  
        {  
            case 0:a=++i;  
            case 1:b=++i;  
            case 2:c=++i;  
            case 3:i++;  
        }
```

```

        printf("%d#%d#%d#%d" ,a,b,c,i);
    }
}

```

4-5 (H10-15)输入 How are you?<回车>后，下列程序的输出结果是\_\_\_\_\_ (3 分)

```

#include <stdio.h>

int main( )
{
    int word;
    char ch;
    word = 0;
    while ((ch = getchar()) != '?') {
        if (ch == ' ') word = 0;
        else if (word == 0) {
            word = 1;
            if (ch <= 'z' && ch >= 'a') ch = ch - 'a' + 'A';
            else if (ch <= 'Z' && ch >= 'A') ch = ch + 'a' - 'A';
        }
        putchar(ch);
    }
}

```

4-6 (A00-41)已知字符 a 和 0 的 ascii 码值分别是 97 和 48，并设有 int a=97; 则表达式'b'<a==97<'l' 的值等于\_\_\_\_\_。 (3 分)

5-1 求 100 - 999 之间的水仙花数之和（水仙花数是指一个三位数的各位数字的立方和是这个数本身，如  $153=1^3+5^3+3^3$ ）。

```

#include <stdio.h>

int main() {
    //printf("%d", f(-15, 10000000));
    int d1, d2, d3;
    int n;
    int sum;
    sum = 0;
}

```

```
for (_____) {
    for (_____) {
        for (_____) {
            n = d1 * 100 + d2 * 10 + d3;
            if (_____) {
                sum += n;
            }
        }
    }
}

printf("%d\n", sum);
return 0;
}
```

5-2 计算并输出一个非零整数序列(序列非空)中偶数的平均值。所有输入数据用空格隔开，用 0 结束输入。输出数据保留 2 位小数。

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

```
int main() {
    int number;
    int sum, n;
    double average;
    n = 0;
    sum = 0;
    scanf("%d", &number);
    while (_____) {
        if (number % 2 == 0) {
            sum += number;
            n++;
        }
        _____;
    }
    average = _____;
```

```
printf("%.2lf\n", average);
return 0;
}
```

## 5-3

输入一批以某个\*负整数结束的非负整数序列，求其中的偶数之和。

```
#include <stdio.h>

int main(void){
    int x, sum;
    sum = 0;
    scanf("%d", &x);
    while (_____) {
        if (_____) sum = sum + x;
        _____;
    }
    printf("sum=%d\n", sum);
    return 0;
}
```

5-4 输入两个复数的实部与虚部，求出并输出它们的和。要求输出数据保留小数点后两位。

```
#include <stdio.h>

int main(){
    double r1, i1, r2, i2, r3, i3;
    int T;
    scanf("%d", &T);
    while (T--) {
        scanf("%lf%lf%lf%lf", &r1, &i1, &r2, &i2);
        r3 = r1 + r2;
        _____;
        if (r3 == 0) {
            if (_____) {
                printf("0\n");
            } else {
```

```
        printf("%.2lf\n", i3);
    }
} else {
    if (i3 > 0) {
        printf("%.2lf+%.2lf\n", r3, i3);
    } else if (_____) {
        printf("%.2lf-%.2lf\n", r3, i3);
    } else {
        printf("%.2lf\n", r3);
    }
}
}
return 0;
}
```

#### 6-1 统计个位数字(10 分)

本题要求实现一个函数，可统计任一整数中某个位数出现的次数。例如-21252 中，2 出现了 3 次，则该函数应该返回 3。

##### 函数接口定义：

```
int Count_Digit ( const int N, const int D );
```

其中 N 和 D 都是用户传入的参数。N 的值不超过 int 的范围; D 是[0,9]区间内的个位数。函数须返回 N 中 D 出现的次数。

##### 裁判测试程序样例：

```
#include <stdio.h>

int Count_Digit ( const int N, const int D );

int main()
{
    int N,D;
    scanf("%d %d", &N, &D);
    printf("%d\n", Count_Digit(N,D));
    return 0;
}
```

/\*你的代码将被嵌在这里\*/

输入样例:

-21252 2

输出样例:

3

### 7-1 40050-1 编程

求不小于给定正整数的最小素数，这个素数还应该是 $(4k+1)$ 的形式(存在某个非负整数  $k$ )。比如素数  $13=(4*3+1)$ 。

输入格式:

在一行中输入一个不超过 99999999 的正整数。

输出格式:

输出大于等于该数的最小且形为 $(4k+1)$ 的素数。

输入样例:

18

输出样例:

29

7-2 (10074)数字金字塔(10 分)

输入正整数  $n$  ( $0 < n < 10$ ), 输出 1 个  $n$  层数字金字塔。

**输入格式:**

在一行中输入一个正整数  $n$  ( $0 < n < 10$ )。

**输出格式:**

输出一个数字金字塔, 每个数字以字符方式打印。

**输入样例:**

在这里给出一组输入。例如:

8

**输出样例:**

在这里给出相应的输出。例如:

1

222

33333

4444444

555555555

66666666666

7777777777777

888888888888888



## 2020-2021 学年第一学期期中考试试卷参考答案

1-1

【正解】F

【解析】 $3 < x < 5$  应表示为  $3 < x \ \&\& \ x < 5$ 

【考点延伸】《考试宝典》2.2、运算符和表达式

1-2

【正解】T

【解析】C 语言中， $\%x$  的意思是以十六进制数形式输出整数， $\%d$  是十进制整数， $\%c$  是单个字符。字符 '1' 的 ASCII 码是 49，因此十六进制输出为 31，十进制输出为 49，按单个字符即为 '1'。

【考点延伸】《考试宝典》2.1、常量和变量

1-3

【正解】F

【解析】 $!6$  是逻辑非运算，所以  $!6=0$ ， $!(6)=1$ 。

【考点延伸】《考试宝典》2.2、运算符和表达式

1-4

【正解】F

【解析】赋值表达式的返回值为赋值符号右边的值，所以当执行 for 循环体中的  $k=1$  语句时，返回值恒为 1，循环一直进行，不会中止。

【考点延伸】《考试宝典》4.3、for 语句

1-5

【正解】T

【解析】形参也是局部变量。

【考点延伸】《考试宝典》8.1、函数的定义与声明

1-6

【正解】T

【解析】 $\text{if}(x=5)$  的返回值恒为 5，即为 True，将执行 y 自增，不会有编译错误。

【考点延伸】《考试宝典》2.2、运算符和表达式

4-1

【正解】3#6#9#

【解析】main 函数中循环体共执行 3 次。fun() 函数中，a 是用 static 声明的静态局部变量，在函数调用结束后不消失而保留原值。三次调用 fun() 函数结束时，a 的值分别为 3,6,9。所以结果为 3#6#9#。

【考点延伸】《考试宝典》7.1、变量的作用域

4-2

【正解】40

【解析】i 递减，j 每次加上 i 的值。j 加的最后一个是 6，i 执行自减后为 5，不再满足  $i > 5$ ，遂跳出循环体。j 的最终值为  $10+9+8+7+6=40$ 。

【考点延伸】《考试宝典》4.3、for 语句

4-3

【正解】7#0#1

【解析】注意 if 语句后面没有花括号，将只会执行一次赋值运算。整理代码为如下格式。

if (a &lt; b) t = a;

a = b;

b = t;

if (a &lt; c) t = a;

```
a = c;  
c = t;
```

结合初值可得 a, b, c 的最终值为 7, 0, 1。

【考点延伸】《考试宝典》3.2、if 语句

4-4

【正解】1#2#3#4

【解析】循环体只会执行一次。i 初始值为 0, 满足 case:0。i 自增为 1 后将 1 赋给 a, 因此又满足了 case:1, 将 2 赋给 b, 以此类推。最终 a,b,c,i 的值分别为 1,2,3,4, 同时跳出循环。

【考点延伸】《考试宝典》3.3、switch 语句

4-5

【正解】how Are You

【解析】这段程序的输入以问号结束（不包括问号字符），目的是将首字母和每一个空格后的字符大小写互换。word 变量的初始值为 0, 每次扫描到空格时也会为 0。当 word 变量的值为 0 时, 表示下一个字符需要进行大小写转换。进行大小写转换时, 先判断其是大写字母还是小写字母, 再与'a'和'A'作差作和即可。

【考点延伸】《考试宝典》2.1、常量和变量

4-6

【正解】1

【解析】'b'的 ASCII 码为 98, 'b'<a 表达式值为 0。'l'的 ASCII 码为 49, 97<'l'表达式值为 0。因此表达式 0==0 的值为 1。

【考点延伸】《考试宝典》2.1、常量和变量

5-1

【正解】d1 = 1; d1 <= 9; d1++

d2 = 0; d2 <= 9; d2++

d3 = 0; d3 <= 9; d3++

n == d1 \* d1 \* d1 + d2 \* d2 \* d2 + d3 \* d3 \* d3

【解析】很明显, d1, d2, d3 分别指这个三位数的百位、十位、个位。按照题目描述循环三位数字即可, 最后的判断条件即是三位数的各位数字的立方和是这个数本身, 如果满足条件, 就累加该数。循环结束后输出结果。

【考点延伸】《考试宝典》4.3、for 语句 3.2、if 语句

5-2

【正解】number != 0

scanf("%d", &number);

1.0 \* sum / n

【解析】因为用 0 结束输入, 所以第一个空应该填入循环条件, 即输入的 number 不为 0。第二空中应为继续读入一个整数, 然后程序会继续执行循环。第三空即是计算偶数的平均值, 用偶数和除以偶数数量即可, 需要考虑到类型转换。

【考点延伸】《考试宝典》4.1、while 语句 1.3、格式化输出/输入函数

5-3

【正解】x >= 0

x % 2 == 0

scanf("%d", &x);

【解析】第一个空是判断输入是否非负, 如果输入为负整数就停止循环。第二个空是判断该整数是否为偶数, 偶数对 2 取余的结果为 0。第三空应为继续读入一个整数, 然后程序会继续执行循环。

【考点延伸】《考试宝典》4.1、while 语句 1.3、格式化输出/输入函数

5-4

**【正解】**  $i3 = i1 + i2$  $i3 == 0$  $i3 < 0$ 

**【解析】**第一个空是计算和的虚部，使用两个输入复数的虚部相加即可。第二空时判断什么时候程序输出为 0，即实部和虚部都为 0 的时候。第三空上下已经有了虚部大于 0 和等于 0 的情况，同时结合输出语句，可以确定条件是虚部小于 0。

**【考点延伸】**《考试宝典》1.3、格式化输出/输入函数 3.2、if 语句

6-1

```
【正解】 int Count_Digit(const int N, const int D) {
    int cnt = 0; //N 中 D 出现的次数
    int n;
    if (N < 0) {
        n = -N; //负号不影响程序结果，如果 N 为负，令 n 为其相反数
    } else n = N; //如果 N 非负，n=N
    while (n != 0) {
        if (n % 10 == D) cnt++; //当前最低位数字等于 D，cnt 累加
        n /= 10; //整除 10，即舍弃最低位
    }
    return cnt;
}
```

**【解析】**见注释**【考点延伸】**《考试宝典》2.2、运算符和表达式

7-1

```
【正解】 int main() {
    int N;
    int M = 1;
    scanf("%d", &N); //读入数据
    for (int k = 0; M <= 99999999; k++, M = 4 * k + 1) { //主循环体
        if (M < N) continue; //所求质数 M 必须大于读入的 N
        int flag = 1; //标志位，1 表示 M 为质数
        for (int i = 2; i < M; i++) { //用 2~M-1 范围内的数去试除 M
            if (M % i == 0) {
                flag = 0; //i 能整除 M，说明 M 不是质数，不必再向下运行
                break;
            }
        }
        if (flag == 1) { //M 是质数，输出结果，结束程序
            printf("%d\n", M);
            break;
        }
    }
    return 0;
}
```

**【考点延伸】**《考试宝典》3.4、对 break 的简单讨论

7-2

```
【正解】int main() {  
    int N;  
    scanf("%d", &N); //读入数据  
    for (int k = 1; k <= N; k++) { //k 为当前输出的层数  
        for (int i = 1; i <= N - k; i++) printf(" "); //输出字符前的空格  
        for (int i = 1; i <= 2 * k - 1; i++) printf("%c", k + '0'); //输出字符  
        printf("\n"); //每层输出完毕后换行  
    }  
}
```

【考点延伸】《考试宝典》4.3、for 语句