

C 语言复习资料

1. C 语言中，变量为何要先定义后使用？

凡是未被事先定义的不作为变量名，这就能保证程序中变量名使用得正确，便于编译时发现错误，避免变量名使用时出错。每个变量被指定一确定类型，在编译时就能为其分配相应的存储单元。指定每一个变量属于一个类型，便于在编译时据此检查该变量所进行的运算是否合法。

2. 举例说明在 C 语言中如何定义符号常量，使用符号常量的优点？

定义：在程序运行过程中，其值不能被改变的量称为常量。

优点：常量可以提高程序的可读性，便于程序的调试和修改。

3. C 语言中，什么基本数据类型不能做 switch () 的参数？

在 C 语言中，float、double 和 long double 等浮点型数据类型不能作为 switch() 的参数

4. 什么是 C 语言的注释方式？

两种方式，1 单行注释 格式：以//开头，从//开始直到本行结束的内容都被视为注释内容。

2 多行注释 格式：以/*开始，以*/结束，/*和*/之间的所有内容都是注释内容。

5. C 语言中的基本数据类型有哪些？

整数型，浮点型，字符型，

6. C 语言中如何进行输入输出操作？

在 C 语言中，输入输出操作主要通过标准库函数来实现，其中最常用的是 `stdio.h` 头文件中定义的函数。

7. 简述 C 语言中函数的作用。

C 语言中函数的主要作用是封装代码块以实现特定功能，提高代码的模块化、重用性和可维护性。通过函数，程序可以更加结构化，易于理解和管理。同时，函数还允许参数化处理和返回值，增加了程序的灵活性和功能性。

8. 如何在 C 语言中声明一个整数变量？

在 C 语言中声明一个整数变量非常简单，你需要使用 `int` 关键字后跟你想要命名的变量名。

9. 解释什么是变量，并给出一个 C 语言中声明变量的例子。

变量是程序中用于存储数据的存储位置。每个变量都有一个特定的名称（变量名），一个数据类型，以及一个存储位置。变量名用于在程序中引用该变量，数据类型决定了可以存储在变量中的数据类型和大小，而存储位置则是计算机内存中的一个位置，用于实际保存数据。

例子：

```
#include <stdio.h>

int main()
{
    // 声明一个整型变量名为 `counter` 并初始化为 0
    int counter = 0;
```

```

        // 声明一个浮点型变量名为 `price` 并初始化为 9.99
float price = 9.99f;

        // 声明一个字符型变量名为 `initial` 并初始化为 'A'
char initial = 'A';

        // 声明一个字符串变量名为 `greeting`
char greeting[] = "Hello, World!";

        // 打印变量值
printf("Counter: %d\n", counter);

printf("Price: %.2f\n", price);

printf("Initial: %c\n", initial);

printf("Greeting: %s\n", greeting);

return 0;
}

```

10. C 语言中，变量为何要先定义后使用？

凡是未被事先定义的不作为变量名，这就能保证程序中变量名使用得正确，便于编译时发现错误，避免变量名使用时出错。每个变量被指定一确定类型，在编译时就能为其分配相应的存储单元。指定每一个变量属于一个类型，便于在编译时据此检查该变量所进行的运算是否合法。

11. While{}和 do{} while 两种循环结构的相同点和不同点是什么？

相同点：1. 循环控制

2. 条件检查
3. 代码块执行
4. 中断和跳转

不同点：1. 条件检查时机

2. 至少执行次数
3. 语法结构
4. 适用场景

12. 什么是 C 语言的运算符？

C 语言的运算符是用于执行数学和逻辑操作的符号。它们用于对操作数（变量、常量或表达式）进行计算，并产生结果。C 语言提供了多种类型的运算符，每种运算符都有特定的用途和优先级。

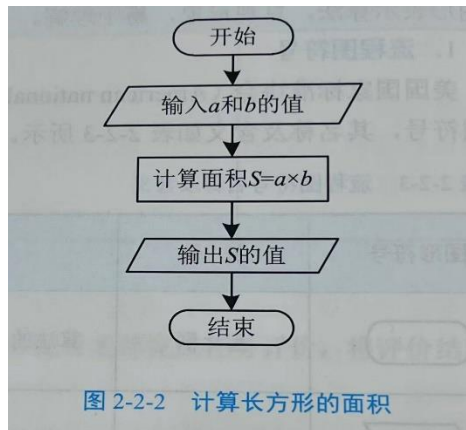
13. 什么是 c 语言的预处理器？

C 语言的预处理器（Preprocessor）是 C 编译器的一部分，它在编译过程的第一阶段运行，处理源代码文件中的预处理指令。

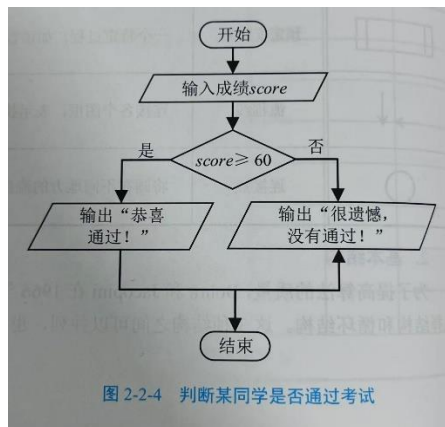
14. 判定是否是闰年算法的表示（流程图）？

开始	->	输入年份 (year)	->	判断year能否被4 整除
是	->	判断year能否被 100整除	->	若能被100整 除，再判断能否 被400整除
能被400整除	->	输出“是闰年”	->	结束
不能被400整除	->	输出“不是闰年”	->	结束
不能被100整除	->	输出“是闰年”	->	结束
不能被4整除	->	输出“不是闰年”	->	结束

15. 请用流程图表示算法，根据长方形的长和宽，计算其面积。



16. 请用流程图表示算法，输入某同学某门课程成绩，判断该同学是否通过考试，输出判断结果。



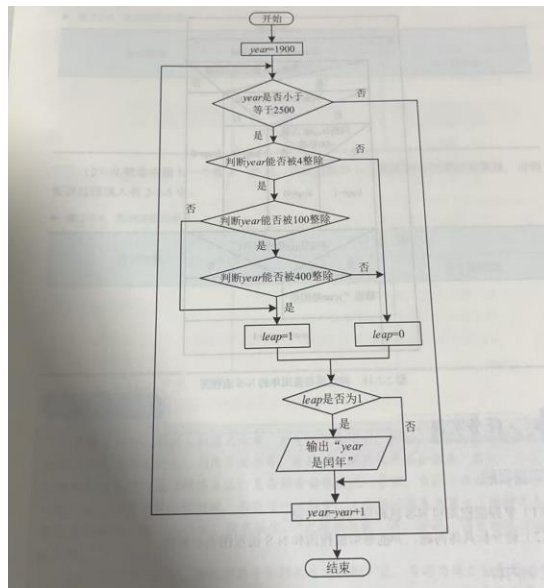
17. 实现计算 1 到 10 的累加和，请画出流程图

开始	->	初始化变量: sum = 0, i = 1	->	判断i是否小于等于10
是	->	sum = sum + i, i = i + 1	->	返回判断i是否小于等于10步骤
否	->	输出sum的值	->	结束

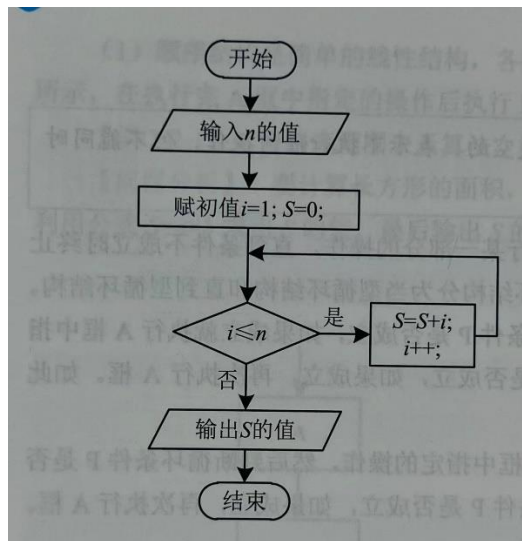
18. 用户输入一个整数，判断并输出其奇偶性，请画出流程图

开始	->	输入一个整数 (num)	->	判断num除以2的余数是否等于0
是	->	输出“该整数是偶数”	->	结束
否	->	输出“该整数是奇数”	->	结束

19. 用流程图表示一个年份是否为闰年,



20. 用流程图表示 $S=1+2+3+..+n$ 的算法。



编程题

1. 编写一个 C 语言程序，判断输入的年份是否为闰年。

```
#include <stdio.h>

int main(){

    int year;

    printf("请输入年份:");

    scanf("%d",&year);

    if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {

        printf("%d 是闰年\n",year);

    } else {

        printf("%d 不是闰年\n",year);

    }

    return 0;

}
```

2. 编写一个 C 语言程序，找出用户输入的三个整数中的最大值。

```
#include <stdio.h>

int main(){

    int num1, num2, num3, max;

    printf("请输入第一个整数:");

    scanf("%d",&num1);

    printf("请输入第二个整数:");
```

```
scanf("%d", &num2);

printf("请输入第三个整数:");

scanf("%d", &num3);

max = num1;

if (num2 > max) {
    max = num2;
}

if (num3 > max) {
    max = num3;
}

printf("这三个整数中的最大值是: %d\n", max);

return 0;

}
```

3. 在屏幕上显示一个等腰三角形。

```
#include <stdio.h>

int main(){

    int i, j, n;

    printf("请输入等腰三角形的行数:");

    scanf("%d", &n);

    for (i = 1; i <= n; i++) {
```



```

        for (j = 1; j <= n - i; j++) {
            printf(" ");
        }
        for (j = 1; j <= 2 * i - 1; j++){
            printf("*");
        }
        printf("\n");
    }
    return 0;
}

```

4. 打印出所有的 “水仙花数”。所谓 “水仙花数” 是指一个 3 位数，其各位数字立方和等于该数本身。例如：153 是一水仙花数，因为 $153=1^3+5^3+3^3$

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

```
int main(){
```

```
    int i, a, b, c;
```

```
    for (i = 100; i < 1000; i++){
```

```
        a = i / 100;
```

```
        b = (i % 100) / 10;
```

```
        c = i % 10;
```

```
        if (i == a * a * a + b * b * b + c * c * c){
```

```
            printf("%d\n", i);
```

```
        }  
    }  
    return 0;  
}
```

5. 输入一个字母 不管大小写都以大写输出

```
#include <stdio.h>  
  
int main(){  
    char ch;  
    printf("请输入一个字母:");  
    scanf("%c",&ch);  
    if (ch >= 'a' && ch <= 'z') {  
        ch = ch - 32;  
    }  
    printf("转换后的大写字母为:%c\n",ch);  
    return 0;  
}
```

6. 使用 switch 输入一个成绩 判断其等级 60 以下差 60-70 中

70-80 良 80-100 优

```
#include <stdio.h>  
  
int main(){  
    int score;  
    printf("请输入成绩:");
```

```
scanf("%d", &score);
switch (score / 10) {
    case 0:
    case 1:
    case 2:
    case 3:
    case 4:
    case 5:
        printf("差\n");
        break;
    case 6:
        printf("中\n");
        break;
    case 7:
        printf("良\n");
        break;
    case 8:
    case 9:
    case 10:
        printf("优\n");
        break;
    default:
```

```
        printf("输入的成绩不合法, 请重新输入正确的  
成绩\n");  
    }  
    return 0;  
}
```

7. 有个人想知道, 一年之内一对兔子能繁殖多少对? 于是就筑了一道围墙把一对兔子关在里面。已知一对兔子每个月可以生一对小兔子, 而一对兔子从出生后第3个月起每月生一对小兔子。假如一年内没有发生死亡现象, 那么, 一对兔子一年内(12个月)能繁殖成多少对?

```
#include <stdio.h>  
  
int main(){  
    int month;  
    int f1 = 1, f2 = 1, f;  
    for (month = 3; month <= 12; month++){  
        f = f1 + f2;  
        f1 = f2;  
        f2 = f;  
    }  
    printf("一对兔子一年内能繁殖成%d 对\n", f2);  
    return 0;  
}
```

8. 编写 九九乘法表

```
#include <stdio.h>

int main(){

    int i, j;

    for (i = 1; i <= 9; i++){

        for (j = 1; j <= i; j++){

            printf("%d * %d = %d\t", j, i, i * j);

        }

        printf("\n");

    }

    return 0;

}
```

9. 百元百鸡 100 元 公鸡 5 元 母鸡 3 元 小鸡 1 元 3 个 公母小 各几只

```
#include <stdio.h>

int main(){

    int x, y, z;

    for (x = 0; x <= 20; x++){ // 公鸡最多买 20 只 (100 元  
全买公鸡, 每只 5 元)

        for (y = 0; y <= 33; y++){ // 母鸡最多买 33 只  
(100 元全买母鸡, 每只 3 元)

            z = 100 - x - y; // 根据总数是 100 只确定小
```

鸡数量

```
        if (z % 3 == 0 && 5 * x + 3 * y + z / 3 == 100) {  
            printf("公鸡:%d 只, 母鸡:%d 只, 小鸡:%d 只  
            \n", x, y, z);  
        }  
    }  
}  
return 0;  
}
```

10.完成等差数列之和

```
#include <stdio.h>  
  
int main(){  
    int a1, d, n;  
    printf("请输入等差数列的首项:");  
    scanf("%d",&a1);  
    printf("请输入等差数列的公差:");  
    scanf("%d",&d);  
    printf("请输入等差数列的项数:");  
    scanf("%d",&n);  
  
    int sum = (n * (a1 + a1 + (n - 1) * d)) / 2;  
    printf("该等差数列的和为:%d\n",sum);
```

```
return 0;
```

```
}
```