

C语言

Q1:素数

```
1  #include <stdio.h>
2  // 判断 x 是否是素数
3  int isprime(int x) {
4      if (x <= 1) return 0; // 0 和 1 不是素数
5      for (int i = 2; i * i <= x; i++) {
6          if (x % i == 0) return 0; // 如果 x 能被 i 整除, 则不是素数
7      }
8      return 1; // 否则是素数
9  }
10 int main(){
11     int x,y;
12     scanf("%d,%d",&x,&y);
13     int cnt = 0; //计数
14     for(int i=x;i<=y;i++){
15         if(isprime(i)){
16             cnt++;
17         }
18     }
19     double p = (double)cnt/(y-x); //强制类型转换
20     printf("%.2lf",p);
21     return 0;
22 }
```

①: isprime函数

②: == 判断是否相等 = 赋值

Q2:循环嵌套

- 外层循环: 控制“行”
- 内层循环: 控制“列”

外层转一圈 → 内层从头跑一遍

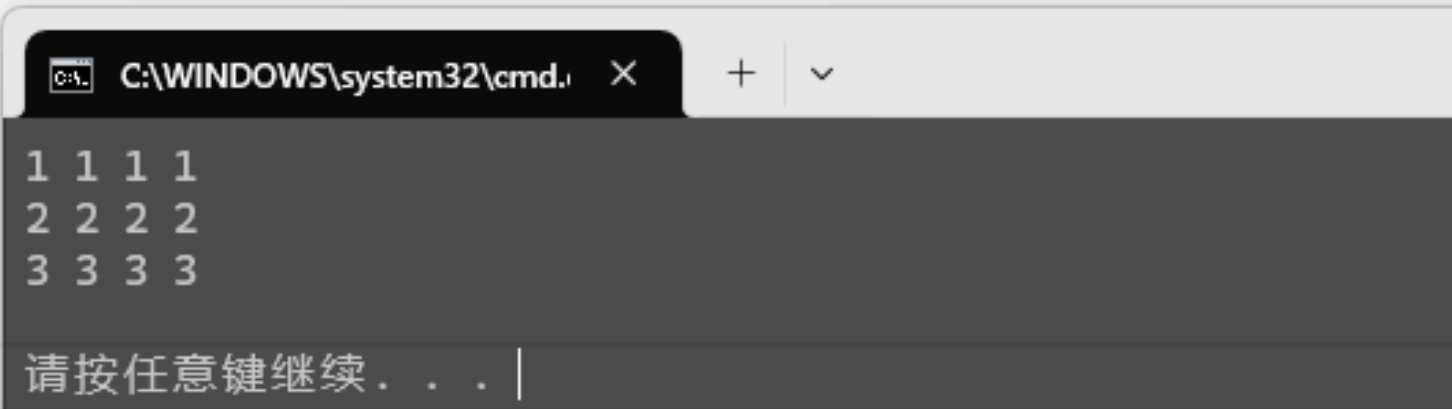
时钟:

- 时针每走 1 小格 (外层)
- 分针就跑一整圈 (内层)

打印矩阵

```
1 1 1 1
2 2 2 2
3 3 3 3
```

```
1 #include <stdio.h>
2
3 int main() {
4     for (int i = 1; i <= 3; i++) {           // 外层循环: 控制行 (row)
5         for (int j = 1; j <= 4; j++) {       // 内层循环: 控制列 (column)
6             printf("%d ", i);               // 打印行号
7         }
8         printf("\n");
9     }
10    return 0;
11 }
12
```



Q3:代码的基本格式

```
1 #include <stdio.h>
2 int main() {
3
4     return 0;
5 }
```

关键字	含义	常用场景
int	有返回值 (返回整数)	函数需要给调用者“一个结果”
void	没有返回值 (什么都不返回)	函数只执行动作, 不需要给结果

Q4:#define

#define 是 C 语言的 宏定义 (macro definition)。
作用：在编译前，把某个名字替换成指定内容。

它不是语句，是 预处理指令
——在编译开始之前就完成替换。

Q5:打印100-200之间的素数

```
1  #include <stdio.h>
2  // 判断 x 是否是素数
3  int isprime(int x) {
4      if (x <= 1) return 0;
5      for (int i = 2; i * i <= x; i++) {
6          if (x % i == 0) return 0;
7      }
8      return 1;
9  }
10 int main() {
11     for (int i = 100; i <= 200; i++) {    // 固定范围: 100~200
12         if (isprime(i)) {
13             printf("%d ", i);            // 打印素数
14         }
15     }
16     return 0;
17 }
```

Q6:打印斐波那契数列

```
1  #include <stdio.h>
2  int main() {
3      int n;
4      scanf("%d", &n);
5      int a = 1, b = 1;    // 前两个数
6      printf("%d %d ", a, b);
7      for (int i = 3; i <= n; i++) {
8          int c = a + b;
9          printf("%d ", c);
10         a = b;
11         b = c;
12     }
13     return 0;
14 }
```

Q7:打印数列 求 π

题目： 用公式 $\pi/4 \approx 1 - 1/3 + 1/5 - 1/7 + \dots$ 求 π 的近似值，直到某一项的绝对值小于 10^{-6} 为止。

```
1  #include <stdio.h>
2  #include <math.h> // 用于fabs函数 (求绝对值)
3
4  int main() {
5      double t = 1.0; // 当前项的值
6      double pi = 0.0; // 累加和 (初始为0)
7      int n = 1; // 分母 (初始为1)
8      int s = 1; // 符号变量 (控制正负交替)
9
10     // 循环条件: 当前项的绝对值 ≥ 1e-6
11     while (fabs(t) >= 1e-6) {
12         pi = pi + t; // 累加当前项
13         n = n + 2; // 分母增加2 (下一项的分母)
14         s = -s; // 符号取反
15         t = (double)s / n; // 计算下一项的值
16     }
17
18     pi = pi * 4; // 最终结果乘以4得到π
19     printf("π的近似值为: %.8f\n", pi); // 输出结果 (保留8位小数)
20
21     return 0;
22 }
```

常用的数学函数：

- ① **pow(x, y)**
- ② **sqrt(x)**
- ③ **fabs(x)** (double 版本) **abs(x)** (int 版本)
- ④ **exp(x)**

常用的头文件：

- #include <stdio.h>** (最常用) 几乎所有程序都需要用它。
- #include <stdlib.h>** 动态内存分配必须用它。
- #include <math.h>** 用数学函数一定要加它。
- #include <string.h>** 操作字符串 (char 数组) 必用。
- #include <ctype.h>** 字符判断、大小写转换必用。