

10.9

③ scanf 扫描集 ○

任何数据类型，
都会先被当作 字符输入
(字符串)

eg: `scanf("%[^\n]", s);`

%[: 开始一个扫描集

^: 表示“除了”(取反)

\n: 换行符

]: 扫描集结束.

⇒ 即读取所有字符
碰到 '\n' 才会停止

eg: `scanf("%[0-9]", s)` ⇒ 只读取数字字符.

`scanf("%[a-zA-Z]", s)` ⇒ 只读取字母字符.

遇到别的, 会立刻停止, 即使后面还有字母, 也不读取.

问题在于缓冲区的 '\n':

eg: 若, `scanf("%[^\n]", s1);`

`scanf("%[^\n]", s2);`

清理

缓冲区

要.

每个 `scanf` 后先跟上一个

`getchar();`

这里 `s2` 就不会被输入任何东西.

因为上一个 `scanf` 末尾有 '\n'.

↓
`getchar` 读取了 '\n' 并将其丢弃.

10.16

① 有时题目给定的数据范围很大,记得用 long long

② 写判断一个数是否为素数(质数)时:

别忘了 `if (num < 1) return 0;`
`if (num == 2) return 1;`

⇒ 0, 1 不是素数
2 是素数 ☆

10.19

③ 同样, 真约数 (除了数本身外,它的因数)

0, 1 没有真约数 ☆

④ "%" 运算符只适用于 整型 数据 ^(int)

⑤ scanf() 的返回值是 成功读取的变量个数

而非读到的具体东西!

若读取失败(格式不匹配)会返回 0.

⑥ 输出 `'\'`, `"`, `;`, `(`, `)` 等字符时,

要在前面加一个 '\' (反斜杠)

eg: 要输出 "ACM!!!\n" 这个整体,

应 `printf ("\\ACM!!!\\n\\");`

10.19

Date

一些初级技巧:

① 逆序存储:

```
for (int i = 0; i < (len / 2); i++) {
    char temp = s[i];
    s[i] = s[len - i - 1];
    s[len - i - 1] = temp;
}
```

" $i < (len / 2)$ " 能提升效率

若 len 为偶数: $len / 2 = 4 / 2 = 2$
 abcd, 成对交换

若 len 为奇数: $len / 2 = 5 / 2 = 2$
 abcde, 由于 $len / 2$ 是整数, 中间 c 不动, 成对交换。

② 判断是否为闰年 函数.

```
int isLeapYear(int year) {
```

// 闰年规则:

// 能被 4 整除但不能被 100 整除, 或

// 能被 400 整除

```
return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
```

}

③ 字母字符, 字母内循环移动($\%26$)

要求: 若是小写字母, 在小写字母内向右移动了, 输出

若是大写字母, 在大写字母内向右移动了, 输出.

不能直接 $s[i] + 3$ (会边界溢出)若小写: 应该 $s[i] = 'a' + (s[i] - 'a' + 3) \% 26$

④

维吉尼亚密码解密公式

① 若密钥key长度较短须循环使用:

$$\text{char } k = \text{key}[i \% (\text{len-key})];$$

② 解密公式: (将A-Z转化为0-25的数字)

$$P = (C - k + 26) \% 26$$

$$\text{明文} = (\text{密文} - \text{密钥} + 26) \% 26$$

知道对应数字后,

用 $\text{plain} = 'A' + \text{num}$ 即可

⑤ fgets() 函数

(读取的最大字符数, 包括结尾的NULL)

原型: $\text{char}^* \text{fgets}(\text{char}^* \text{str}, \text{int } n, \text{FILE}^* \text{stream});$

(指向要被输入的字符串组)

(用 stdin)

特性: 能输入换行符 \n 等看不见的字符

② 自动加 NULL 终止符, 不会溢出

返回值: 成功则返回指针 str, 失败或到达文件末返回 NULL

eg: $\text{char } \text{buf}[10];$
 $\text{fgets}(\text{buf}, \text{sizeof}(\text{buf}), \text{stdin});$

(只能读入9个字符, 会自动+NULL)

10.22

依旧补充

C++中若要表示 $\ln(x)$ 函数
 写成 " $\log(x)$ " 即可
 而原本的 \log 要写成 $\log_{10}(x)$

① 关闭输入输出同步流

② 条件运算符 (条件)? 表达式1: 表达式2;

若条件为真, 整个式子取表达式1的值,
 若条件为假, 整个式子取表达式2的值。

(简单的条件判断推荐这样写, 复杂的嵌套用 if-else 更好。)

10.26

① 递归阶乘函数

```
long long factorial(int n){
    if(n==0 || n==1){
        return 1; // 处理边界情况
    }
    return  $n * \text{factorial}(n-1)$ ; // 递归计算 n!
```

② 约瑟夫环模拟

有 n 个人围成一圈, 轮流依次 1, 2, 3 报数, 报到了的退出, 问最后留下的人是几号。

$a[n] = \{0\}$; // 报到了则 $a[j] = 1$; 一开始都初始为 0。

由于是圆圈循环, 数组下标处理: $j = (j+1) \% n$