1. Single

1）桶计数法

所有数据里最大值是多少（如n），就开一个多大的数组（或n+1,更方便一些），出现了某个数字i，那就让a[i]++表示i这个数字出现了一次。计数完后来个for循环从a[1]到a[n]来读取记录的出现次数，找出a[ j ]=1的j即可。

2）另一种做法：利用异或运算。

for (int i = 0; i < n; i++) {

scanf("%d", &x);

ans = ans ^ x; } 或for (scanf("%d", &x); scanf("%d", &x) != EOF; ans ^= x);（不推荐）

1. Factorial
2. 算法：加减乘法先计算再取余🡪边计算边取余（防止int溢出）
3. 另一种做法：原理：利用阶乘的“递推性”，即n! = (n-1)! \* n.

#include <stdio.h>

const int Mod = 10007; int n, sum = 0; //对常量、核心变量的定义

int main(void) {

scanf("%d", &n);

for (int i = n; i >= 1; i--) sum = (sum + 1) \* i % Mod;

printf("%d", sum);

return 0; }

1. Inverse2

***如何读入字符串 / 字符？（重要！！！）***

我们在此进行简单的归纳。假设已经定义了 str[LEN] 。

1) scanf("%s", str) ：读入一个字符串（与 %d 不同， %s 所对应的 str 前通常不加 & ），但读到空格就会停止。因此本题不能简单地使用它，在输入的字符串没有空格的时候推荐大家使用。

2) for (...) str[i] = getchar() ：读入一个字符，理论上推荐大家使用，但初学者可能难以理解字符系统的显示行为，因此可能会产生与想象不一致的后果。

3) for (...) scanf("%c", &str[i]) ：读入一个字符，基本于 getchar() 等同，但在实际中似乎使用不多。

4) gets(str) （ fgets 等）：读入一行， gets 有历史渊源，不推荐大家使用。 如果在读入n之后直接采用循环 + getchar() / scanf("%c") 的方式来读入数组的话，就会首先读入换行符，因此，需要使用一个 getchar() 或 scanf("%c") 来首先读掉这个换行符，之后才能采用循环读入一个字符的方法读入。

为什么 scanf("\n") 不行？ scanf("\n") 的行为并不只是读一个换行符，事实上，scanf 将 '\n' ,' ' 等字符都视为空白符，格式串里的一个空白符会匹配输入中连续的若干个的空白符，如果下一行中的字符串以空白符开头，就发生了问题。