

ВТОРО КОНТРОЛНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР ВЕЛИКО ТЪРНОВО, 7 МАЙ, 2017 Г. , ГРУПА А

Автор: Антон Анастасов

ЗАДАЧА АК4. ДВОИЧНО ТЪРСЕНЕ

Антон участва в състезания по програмиране от доста време. След една седмица му предстои контролно за определяне на националния отбор. Докато решаваше една от задачите от предишно контролно, му хрумна, че една малка част от нея се свежда до двоично търсене. Като един опитен състезател, той седна и бързо сътвори следната C++ имплементация на двоично търсене:

```
int binary_search(vector<int> &a, int value) {
    int l = 0;
    int r = (int)a.size() - 1;
    while (l <= r) {
        int m = (l + r + 1) / 2;
        if (a[m] == value)
            return m;
        else if (a[m] > value)
            r = m - 1;
        else
            l = m + 1;
    }
    return -1;
}
```

Преди да се заеме да тества имплементацията за правилност, Антон осъзна, че всъщност неговият целочислен вектор a не е сортиран. За щастие, той бързо осъзна как да се справи с този проблем. След това, обаче, си зададе следния въпрос: „Каква е вероятността имплементацията ми на двоично търсене да работи правилно върху целочислен масив, който не е сортиран?“ Тъй като теорията на вероятностите не му е толкова любима, за начало той реши да формулира изникналата пред него задача по следния начин:

Колко е броят на пермутациите на целите числа между 1 и N включително, такива, че `binary_search`, чиято имплементация е дадена по-горе, връща индекс i , за който $a[i]$ е равно на $value$? Числото може да е много голямо, но той се интересува от остатъка му при делене на 1 000 000 007. Помогнете на Антон, като напишете програма **binsearch**, която решава тази задача.

Вход

На единствения ред на входа са зададени две положителни цели числа N и $value$, разделени с интервал. *Гарантира се, че $value$ е между 1 и N , включително.*

Изход

Нека P_N е броят на пермутациите на числата от 1 до N включително, за които имплементацията на двоично търсене на Антон връща индекс, съответен на $value$. На единствения ред на изхода програмата трябва да извежда едно цяло число, равно на остатъка при делението на P_N на 1 000 000 007.

Ограничения

$1 \leq N \leq 10\,000$.

В 10% от тестовете, $1 \leq N \leq 10$.

В 20% от тестовете, $1 \leq N \leq 20$.

В 50% от тестовете, $1 \leq N \leq 100$.

В 75% от тестовете, $1 \leq N \leq 1\,000$.

Ограничение на памет: максимум 16 MiB

**ВТОРО КОНТРОЛНО СЪСТЕЗАНИЕ
НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
ВЕЛИКО ТЪРНОВО, 7 МАЙ, 2017 Г. , ГРУПА А**

Примери

Вход	Изход
3 1	4
5 2	66
7 1	2160
9 4	135360

Обяснение към пример 1:

6-те пермутации на числата $\{1, 2, 3\}$ са $(1, 2, 3)$, $(1, 3, 2)$, $(2, 1, 3)$, $(2, 3, 1)$, $(3, 1, 2)$, $(3, 2, 1)$.

Измежду тези пермутации, двоичното търсене на Антон не работи единствено за две от тях: $(2, 3, 1)$ и $(3, 2, 1)$. Броят на пермутациите, за които двоичното търсене на Антон работи, е $6 - 2 = 4$. Забележете, че остатъкът при делене на $1\,000\,000\,007$ е 4 , което е правилният изход.