

位运算

【故事背景】

JYY 最近在研究位运算。他发现位运算中最有趣的就是异或 (xor) 运算。对于两个数的异或运算, JYY 发现了一个结论: 两个数的异或值为 0 当且仅当他们相等。于是 JYY 又开始思考, 对于 N 个数的异或值会有什么性质呢?

【问题描述】

JYY 想知道, 如果在 0 到 $R-1$ 的范围内, 选出 N 个不同的整数, 并使得这 N 个整数的异或值为 0, 那么一共有多少种选择的方法呢? (选择的不同次序并不作重复统计, 请参见样例)

JYY 是一个计算机科学家, 所以他脑海里的 R 非常非常大。为了能够方便的表达, 如果我们将 R 写成一个 0-1 串, 那么 R 是由一个较短的 0-1 串 S 重复 K 次得到的。比如, 若 $S = "101"$, $K=2$, 那么 R 的 2 进制表示则为 "101101"。

由于计算的结果会非常大, JYY 只需要你告诉他选择的总数对 10^9+7 取模的结果即可。

【输入格式】

从文件 *binary.in* 中读入数据。

第一行包含两个正整数 N 和 K 。

接下来一行包含一个由 0 和 1 组成的字符串 S 。

我们保证 S 的第一个字符一定为 1。

【输出格式】

输出到文件 *binary.out* 中。

输出文件包含一行一个整数, 表示选择的方案数对 10^9+7 取模的值。

【样例输入 1】

```
3 1
100
```

【样例输出 1】

```
1
```

【样例输入 2】

```
5 100
1
```

【样例输出 2】

598192244

【样例说明】

对于第一个样例，唯一的一种选择方法是选择 $\{1, 2, 3\}$ 。

【数据规模与约定】

测试点	N	K	$ S $
1	$N = 3$	$K = 1$	$ S = 10$
2		$K = 10000$	$1 \leq S \leq 50$
3,4,5	$N = 5$	$1 \leq K \leq 20$	
6	$N = 4$	$1 \leq K \leq 10^5$	
7,8,9,10	$3 \leq N \leq 7$		