

5.3 Поиск образца в тексте

Поиск образца в тексте

Найти все вхождения строки *Pattern* в строку *Text*.

Вход. Строки *Pattern* и *Text*.

Выход. Все индексы i строки *Text*, начиная с которых строка *Pattern* входит в *Text*:

$Text[i..i + |Pattern| - 1] = Pattern$.

Реализуйте алгоритм Карпа–Рабина.

Формат входа. Образец *Pattern* и текст *Text*.

Формат выхода. Индексы вхождений строки *Pattern* в строку *Text* в возрастающем порядке, используя индексацию с нуля.

Ограничения. $1 \leq |Pattern| \leq |Text| \leq 5 \cdot 10^5$.

Суммарная длина всех вхождений образца в текста не превосходит 10^8 . Обе строки содержат буквы латинского алфавита.

Пример.

Вход:

```
aba
abacaba
```

Выход:

```
0 4
```

Образец aba входит в позициях 0 (abacaba) и 4 (abacaba) в текст abacaba.

Пример.

Вход:

```
Test
testTesttesT
```

Выход:

```
4
```



Пример.

Вход:

```
aaaaa  
baaaaaa
```

Выход:

```
1 2 3
```

Данный пример демонстрирует, что вхождения могут накладываться друг на друга.

Подсказки по реализации.

- Будьте осторожны с переполнением целого типа. Применяйте операцию взятия по модулю p после каждой арифметической операции.
- Будьте осторожны с взятием отрицательных чисел по модулю.
- Будьте осторожны с операцией взяти подстроки — она может оказаться дорогой по времени и по памяти.