



Санкт-Петербургский государственный  
университет

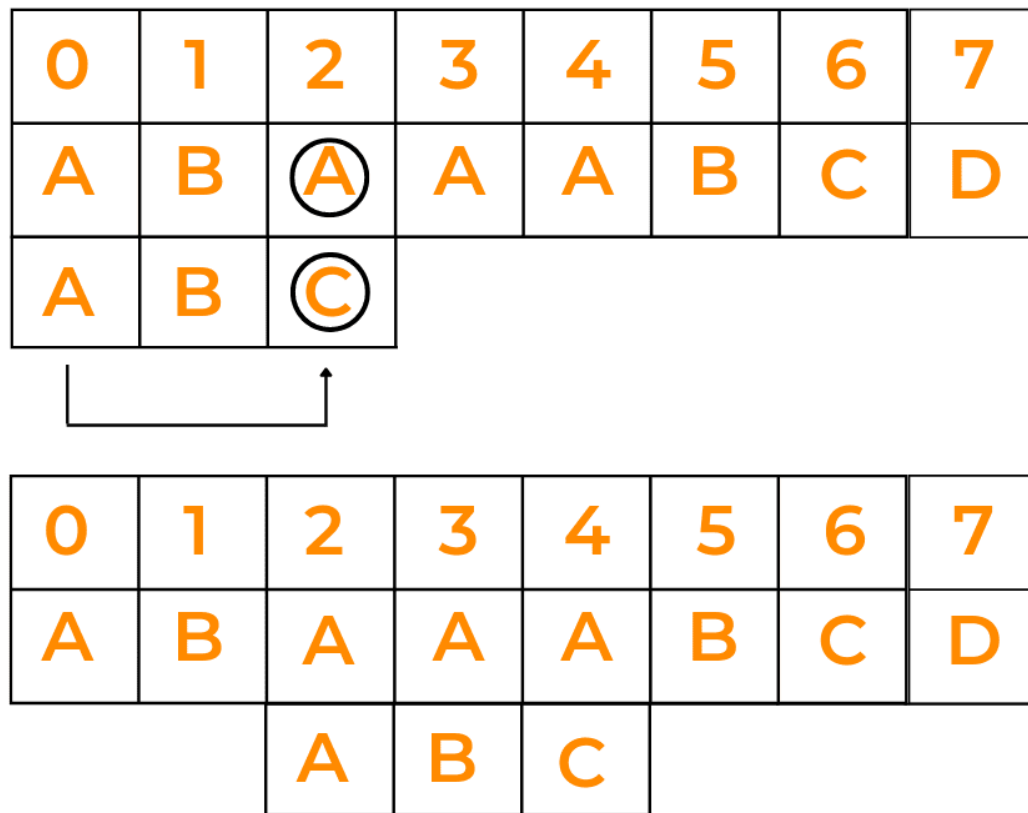
# Алгоритм Бойера – Мура

Макарова Елена Денисовна, группа 23.Б10-мм  
Санкт-Петербург  
2023

# Об алгоритме

- ◆ Предназначен для поиска подстроки в строке
  - ✧ Используется в текстовых редакторах, биоинформатике, сетевой безопасности
- ◆ Роберт Бойер, Джей Мур, 1977 год
- ◆ Наиболее быстрый и эффективный
- ◆ Сравнение символов справа налево, «скачки» через символы, заведомо отсутствующие в шаблоне

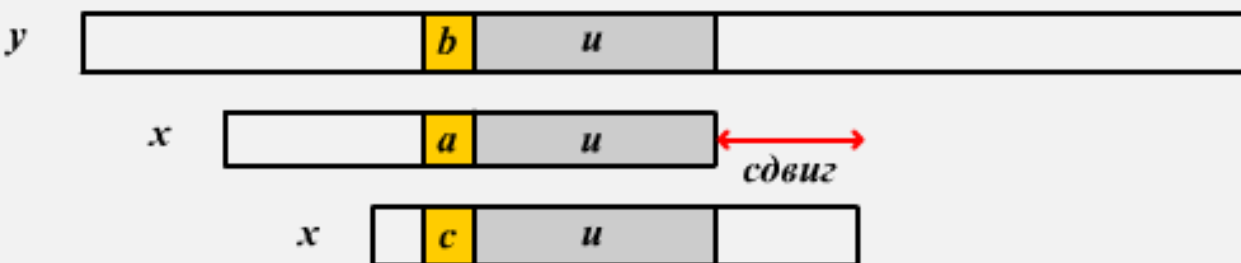
# Об алгоритме



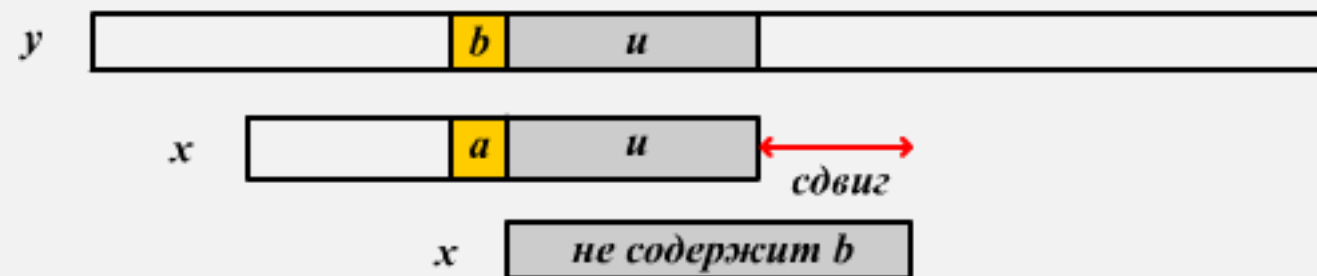
Источник: [favtutor.com](http://favtutor.com)

# Реализация

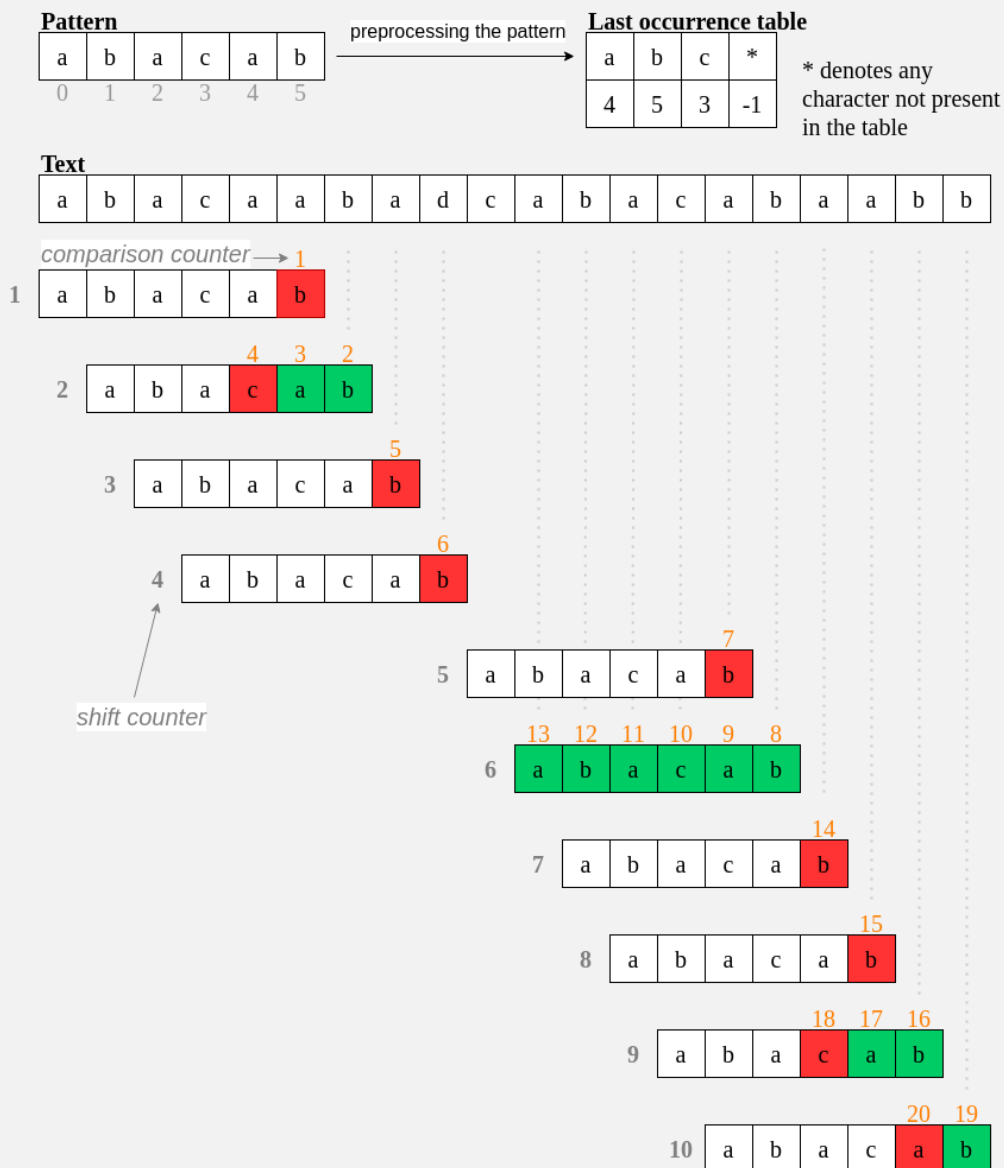
- ♦ Два правила сдвига: «плохой символ» (bad character) и «хороший суффикс» (good suffix)
- ♦ Выбирается максимум между двумя сдвигами
- ♦ Требуют предварительной обработки шаблона



Источник: [neerc.ifmo.ru](http://neerc.ifmo.ru)



# Правило плохого символа



♦ Строится таблица для всех символов алфавита =  $O(|\Sigma|)$  доп. памяти и времени

♦ Выравнивание стоп-символа с его самым правым появлением в шаблоне

# Правило хорошего суффикса

M	A	N	P	A	N	A	M	A	N	A	P	-
A	N	A	M	P	N	A	M	-	-	-	-	-
-	-	-	-	A	N	A	M	P	N	A	M	-

*Источник: en.wikipedia.org*

- ✦ Строится таблица сдвигов для суффиксов подстроки
- ✦ Выравнивание со следующим таким же суффиксом

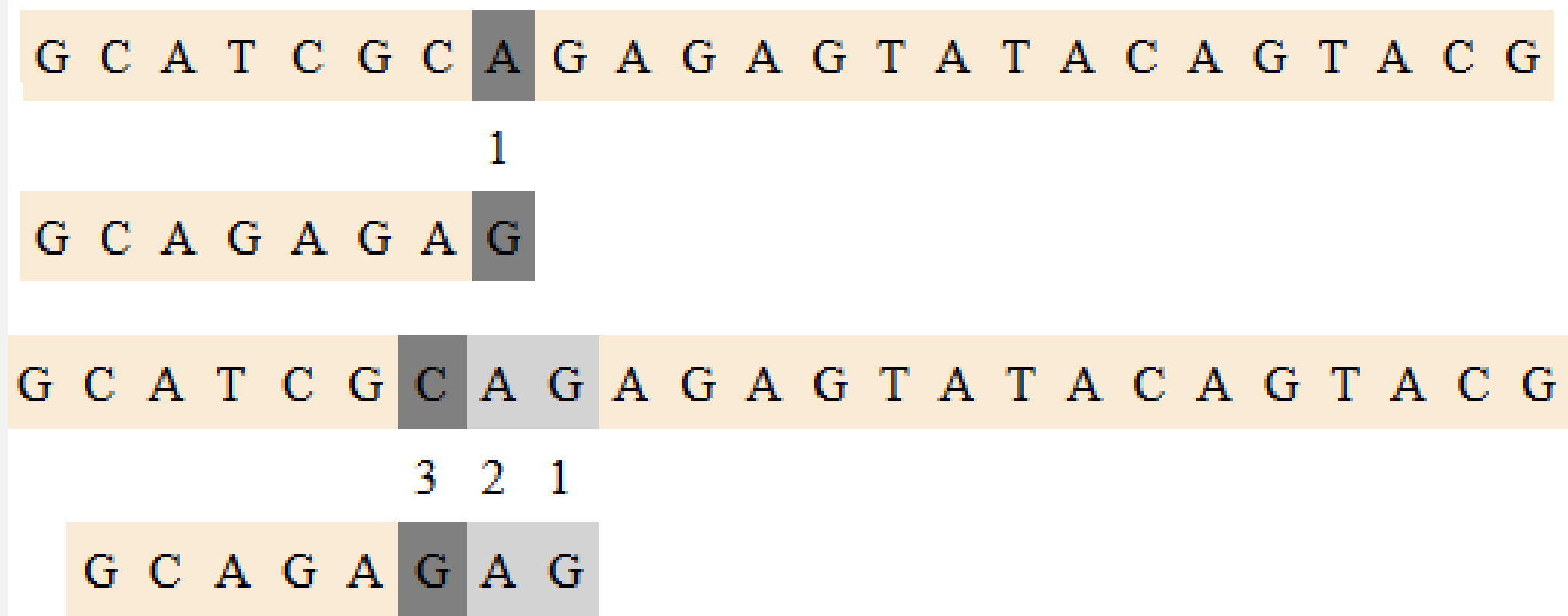
# Пример

$c$	A	C	G	T
$bmBc[c]$	1	6	2	8

$i$	0	1	2	3	4	5	6	7
$x[i]$	G	C	A	G	A	G	A	G
$suff[i]$	1	0	0	2	0	4	0	8
$bmGs[i]$	7	7	7	2	7	4	7	1

Источник: [www-igm.univ-mlv.fr](http://www-igm.univ-mlv.fr)

# Пример



Источник: [www-igm.univ-mlv.fr](http://www-igm.univ-mlv.fr)



# Пример

G C A T C G C A G A G A G T A T A C A G T A C G

8 7 6 5 4 3 2 1

G C A G A G A G

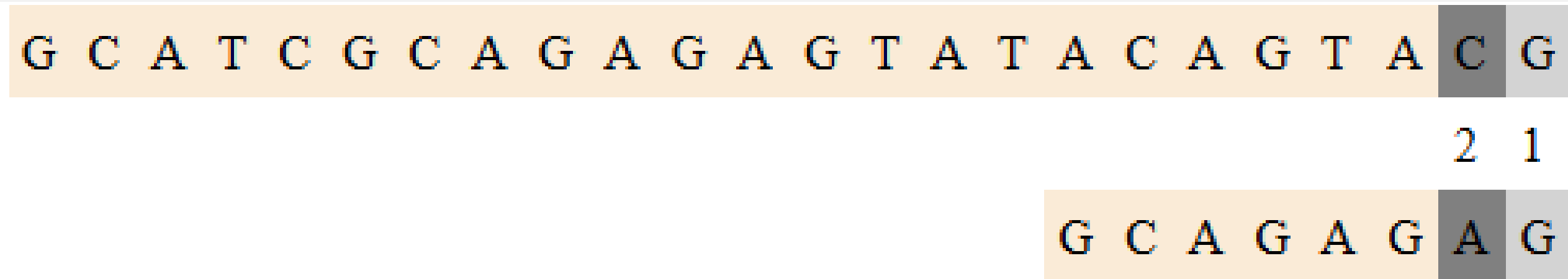
G C A T C G C A G A G A G T A T A C A G T A C G

3 2 1

G C A G A G A G

Источник: [www-igm.univ-mlv.fr](http://www-igm.univ-mlv.fr)

# Пример



*Источник: [www-igm.univ-mlv.fr](http://www-igm.univ-mlv.fr)*

- ◆ Предварительные вычисления требуют  $O(m^2 + |\Sigma|)$  времени и памяти
- ◆ В худшем случае –  $O(mn)$ , в лучшем –  $O(n/m)$

# Достоинства и недостатки

+:

- ✦ Высокая скорость на больших алфавитах
- ✦ Работает быстрее при увеличении длины шаблона
- ✦ Позволяет множество вариаций и модификаций

# Достоинства и недостатки

-:

- ◆ Не расширяется до приблизительного поиска, поиска любой строки из нескольких
- ◆ Расход памяти
- ◆ Существенное снижение скорости на неудачных текстах

# Вариации

- ◆ Алгоритм Бойера – Мура – Хорспула
- ◆ Алгоритм Чжу – Такаоки
- ◆ Алгоритм Апостолико – Крошмора
- ◆ Алгоритм Райта