

# Биоинформатика



## Сборка de-novo

Герасимов Евгений Сергеевич

Many revolutions look  
irrelevant just before they  
change everything swiftly

George Church

# Что делать с данными NGS?

## Секвенирование *de-novo*

Новые геномы (WGS)

Новые транскриптомы (RNA-seq)

## *Ресеквенирование*

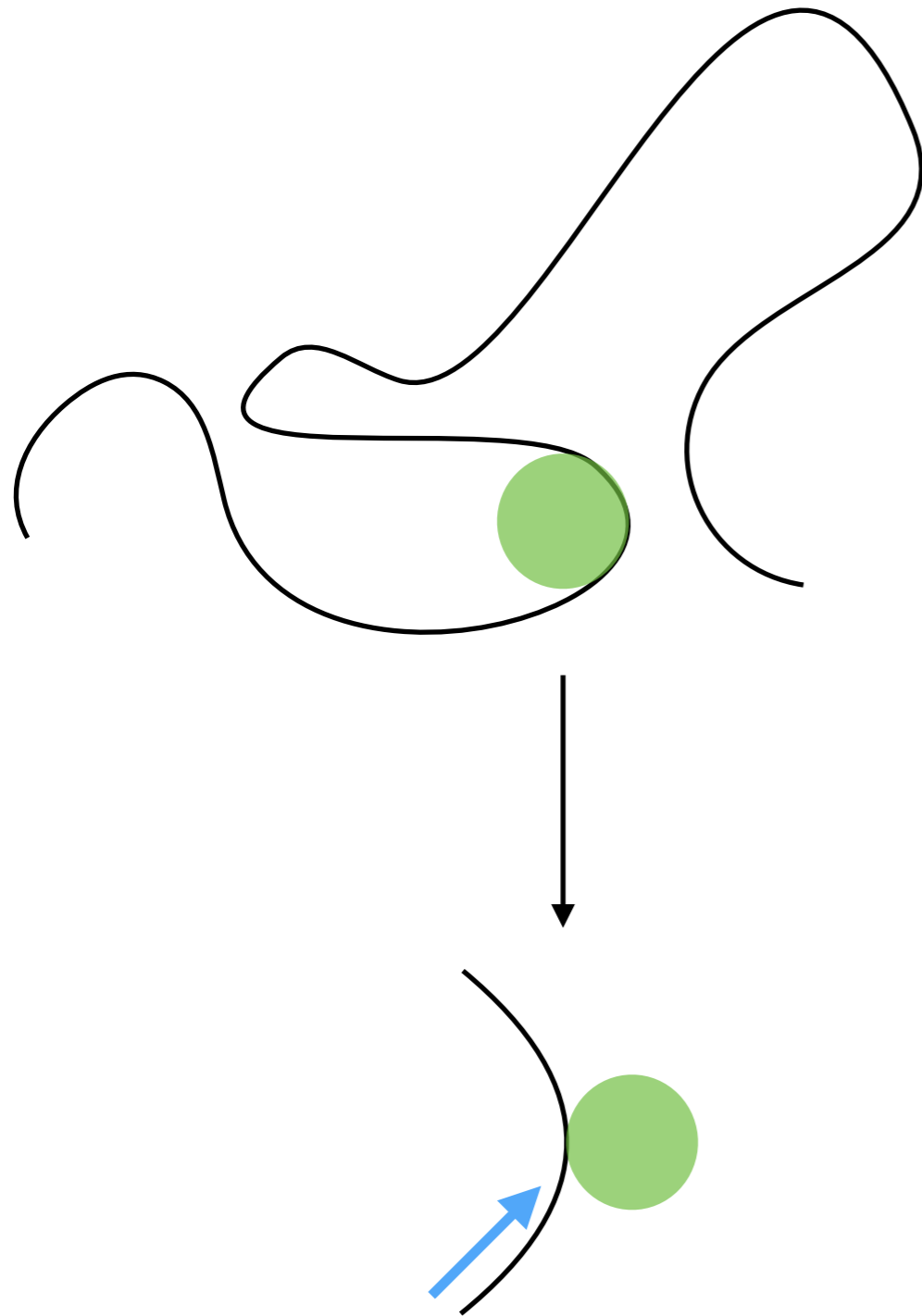
Целевое ресеквенирование

Exome-seq, polyA-RNA-seq, ChIP-seq,  
HiC, NET-seq, MNase-seq

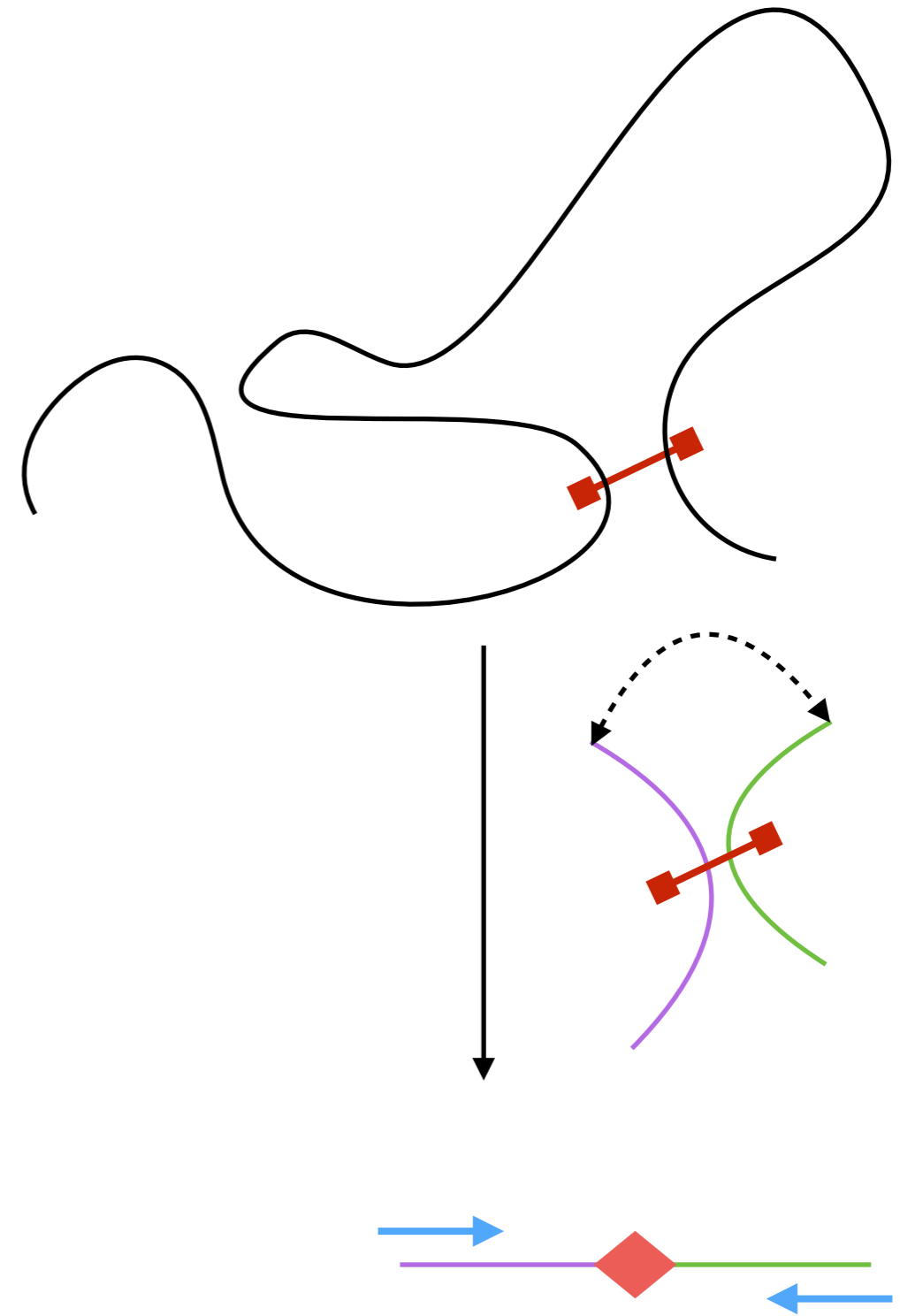
Тотальное ресеквенирование

WGS, RNA-seq

# Принцип целевого секвенирования



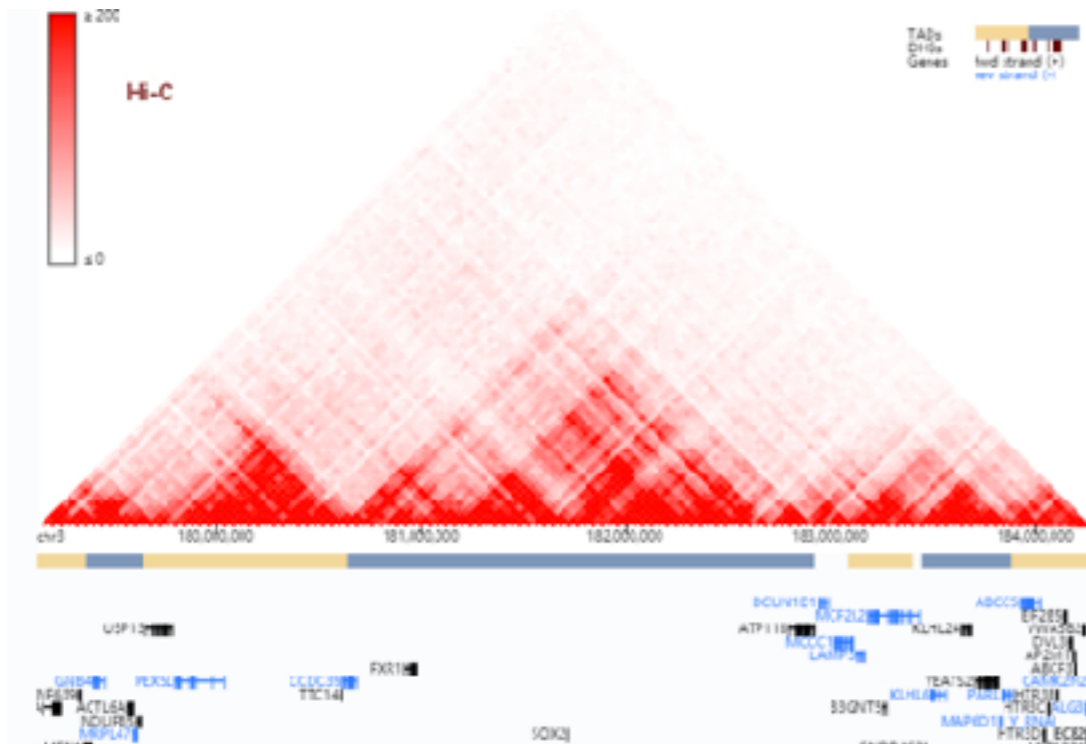
ChIP-seq



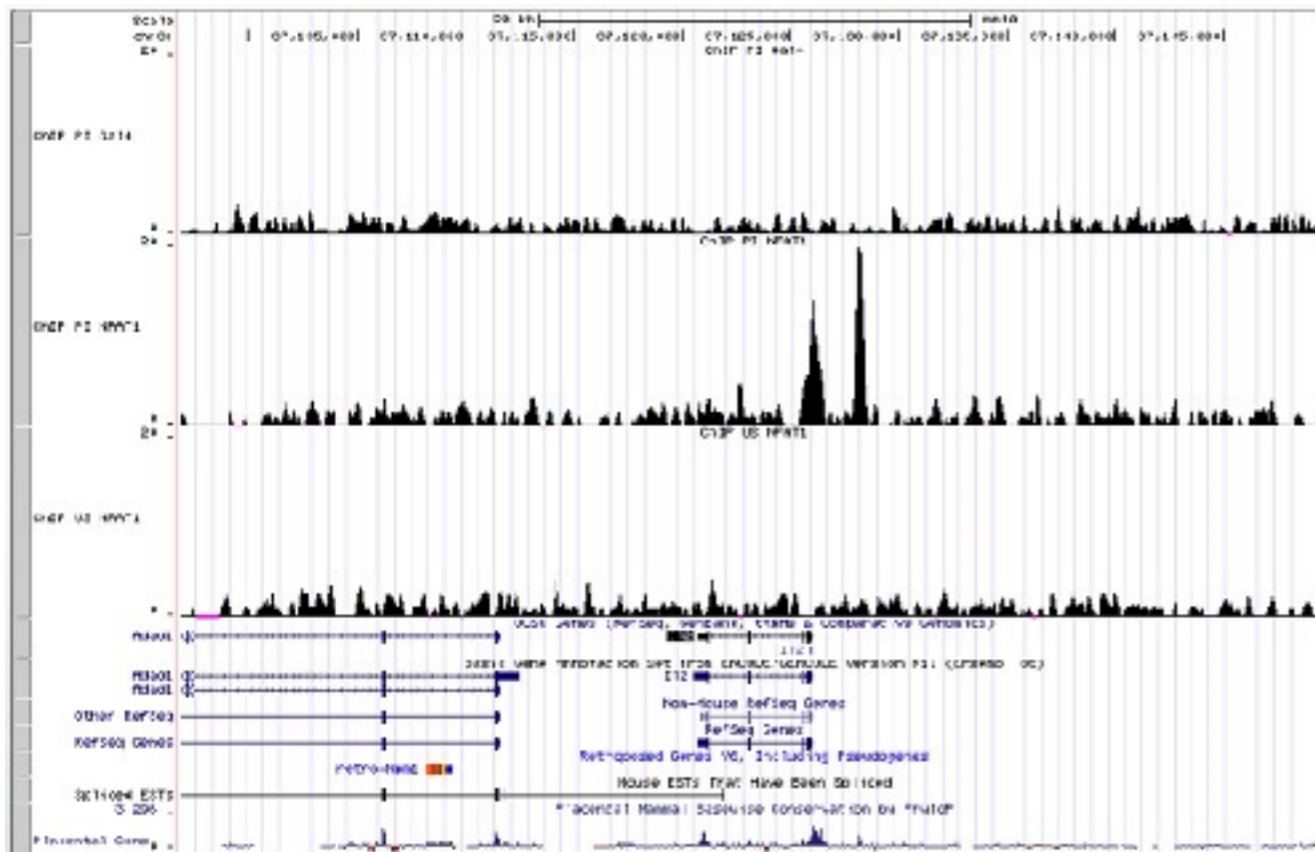
HiC

# Применение NGS

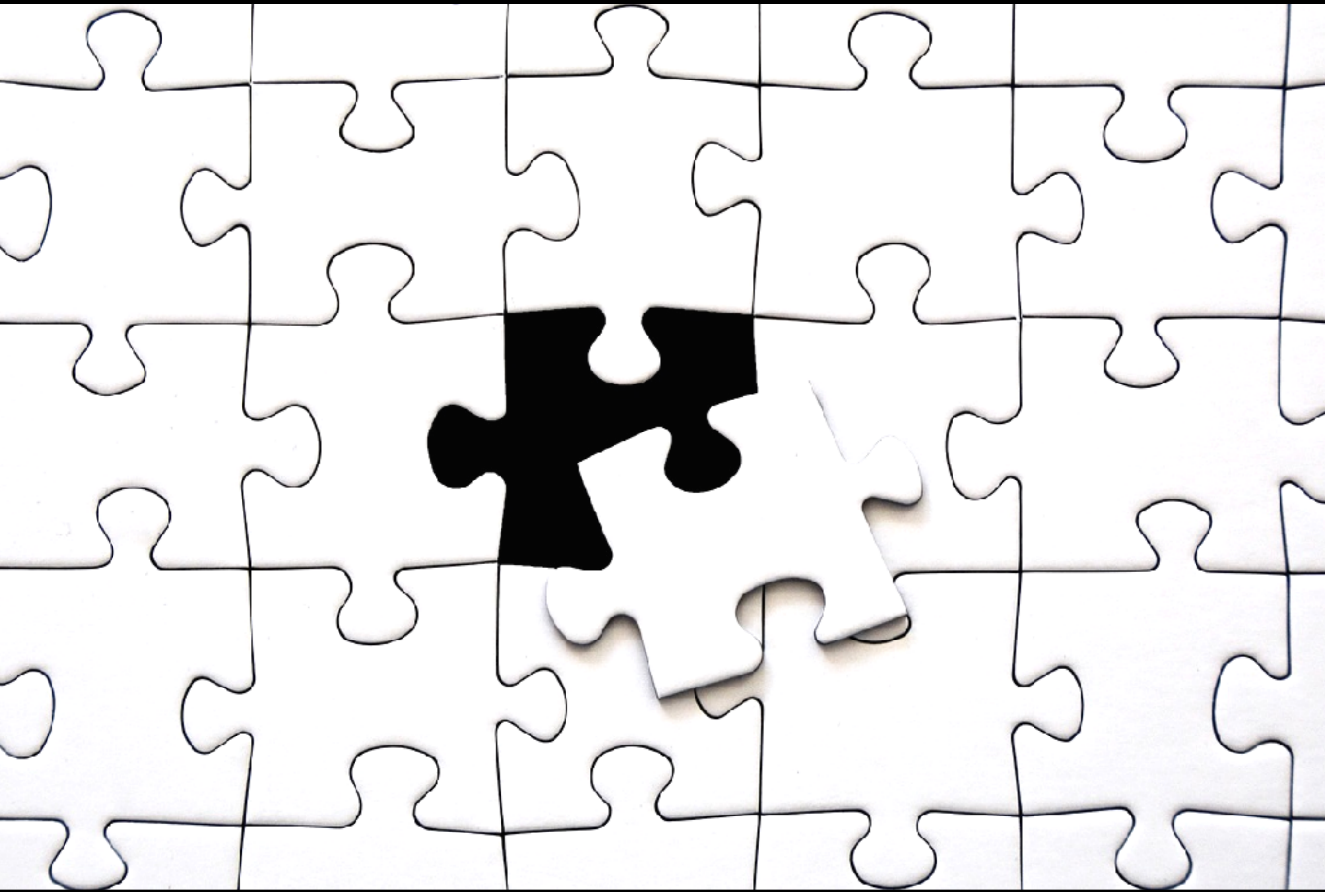
HiC, *карта контактов*



ChIP-seq, *места связывания ТФ*



# СБОРКА



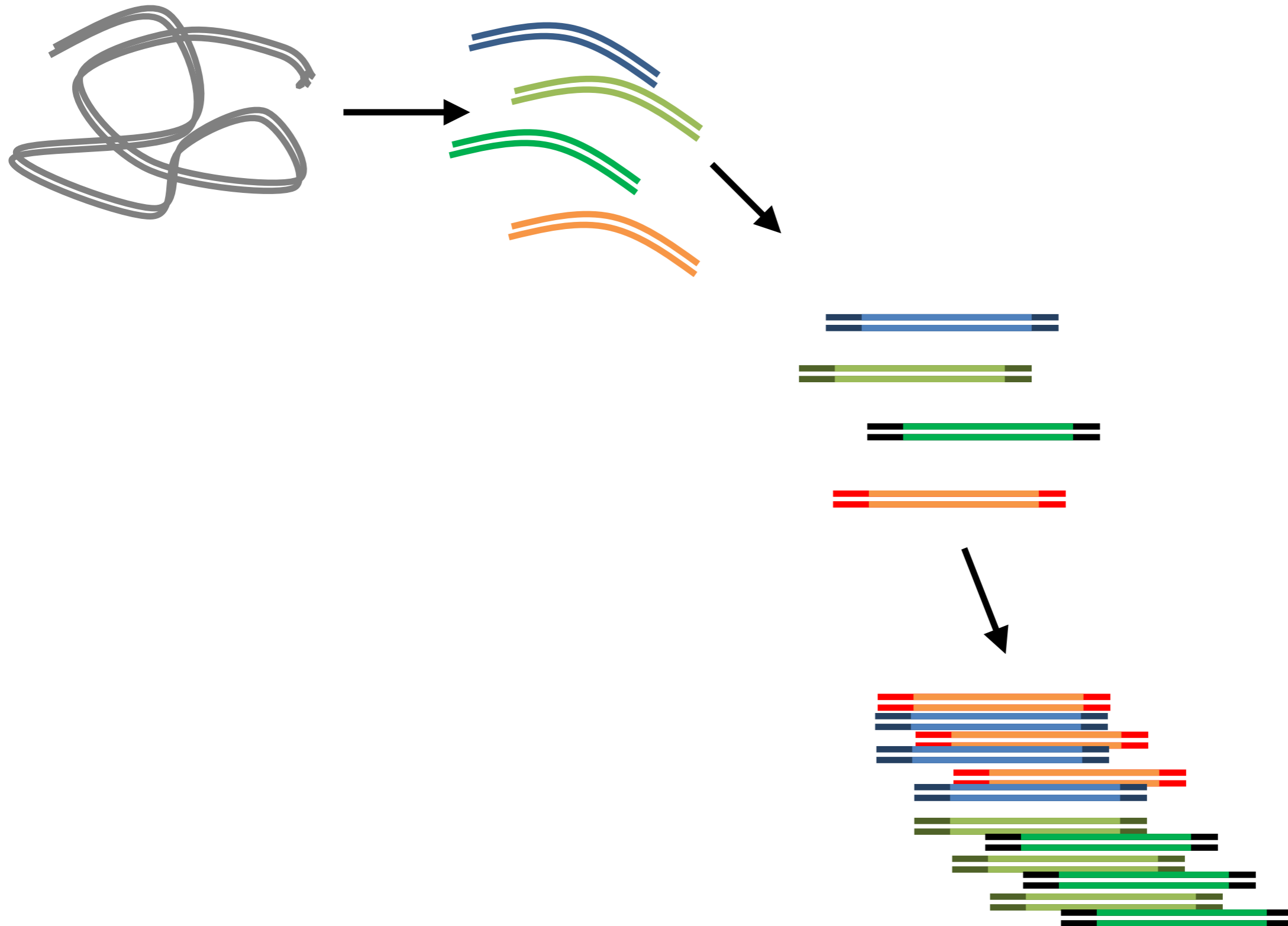
# Пролог

"Каждый биоинформатик хотя бы раз в жизни написал свой сборщик или хотя бы картировщик."

(автор неизвестен)

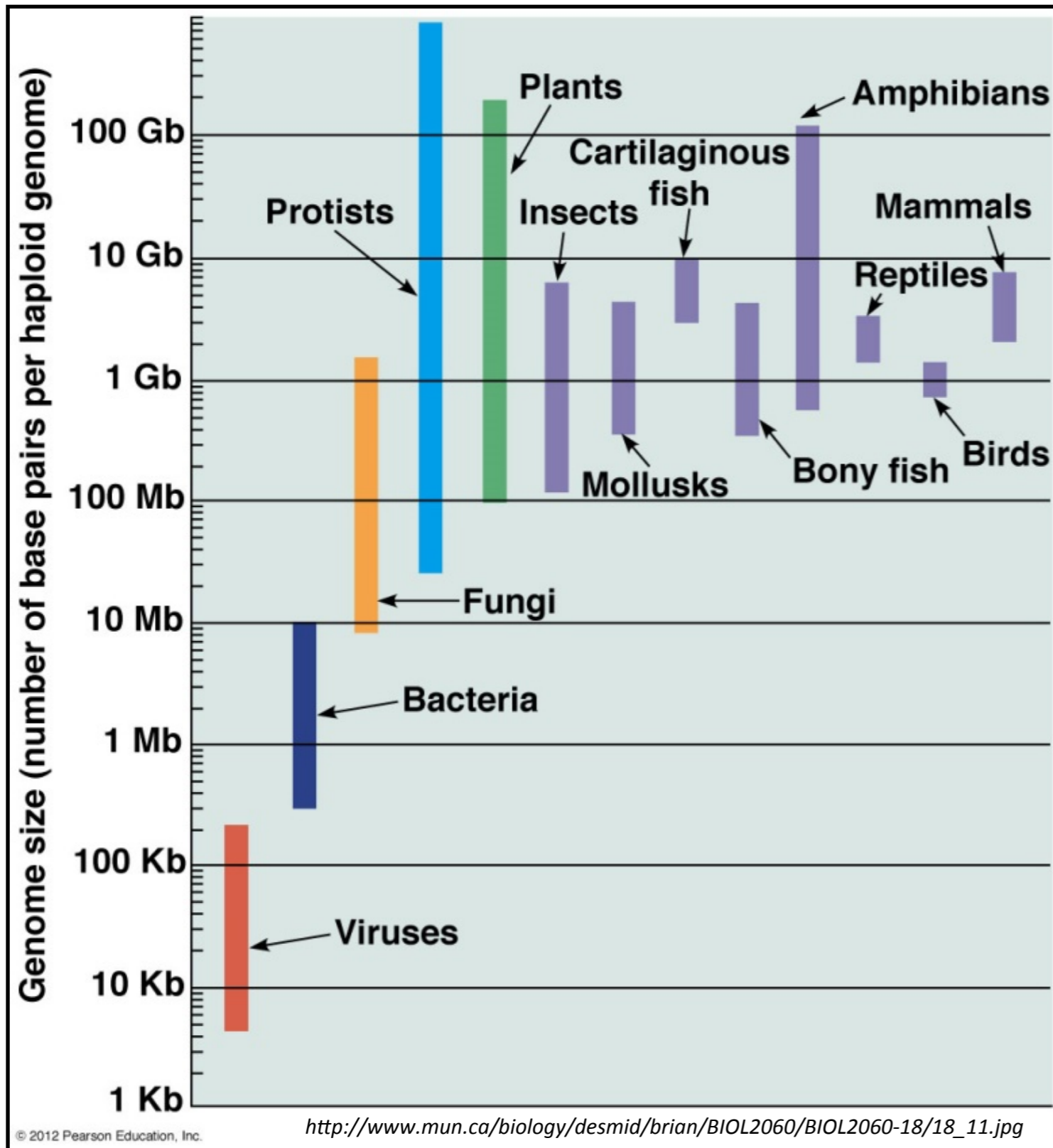
# Принципы NGS (еще раз)

Фрагментация, амплификация, отбор





# Какими бывают геномы



Малые геномы:

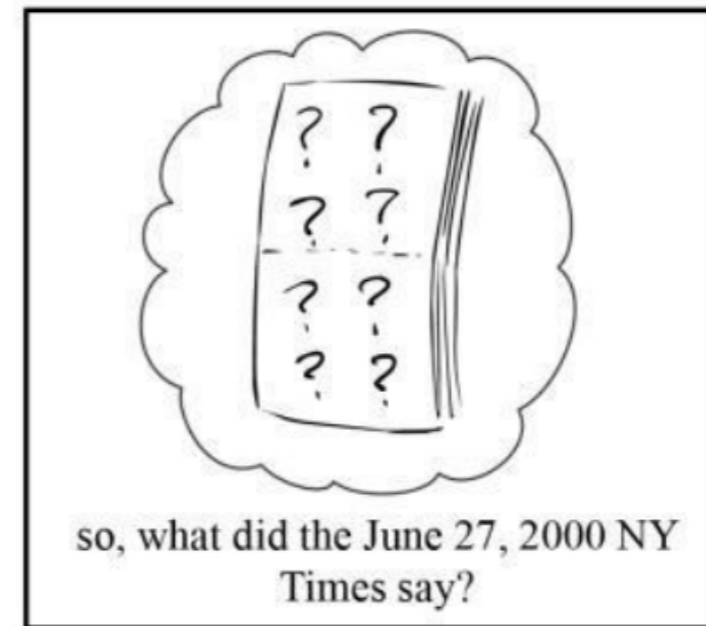
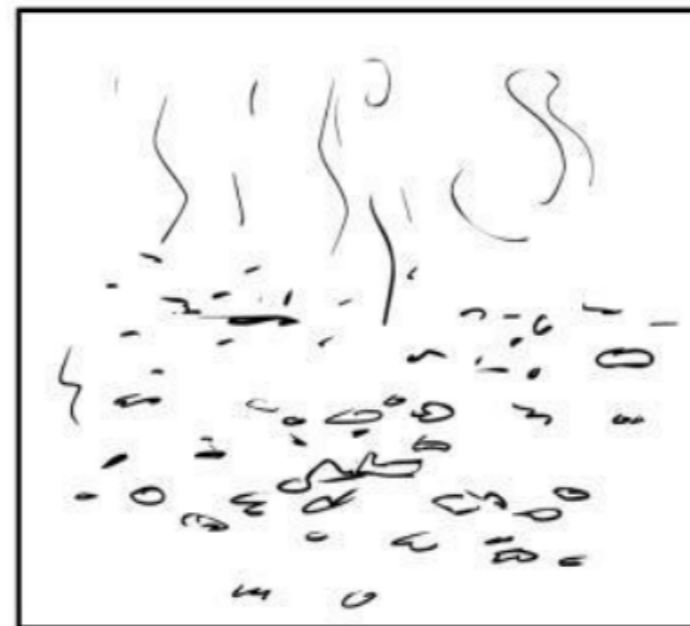
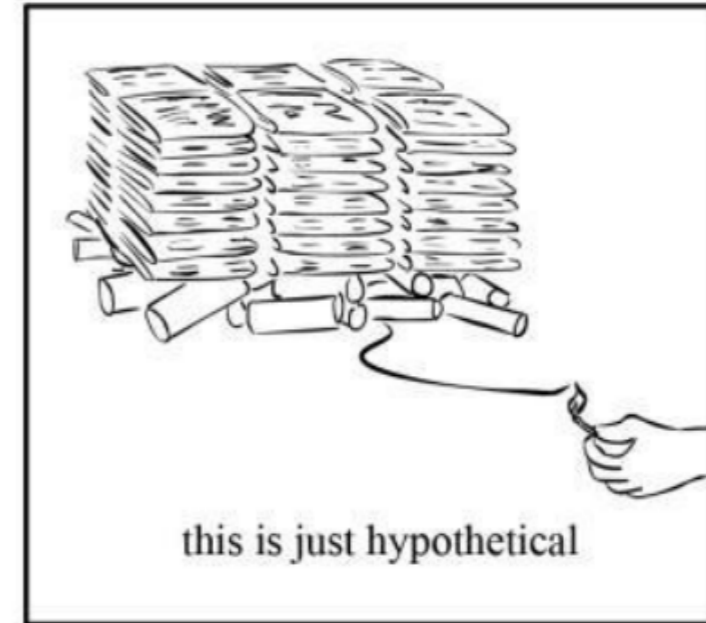
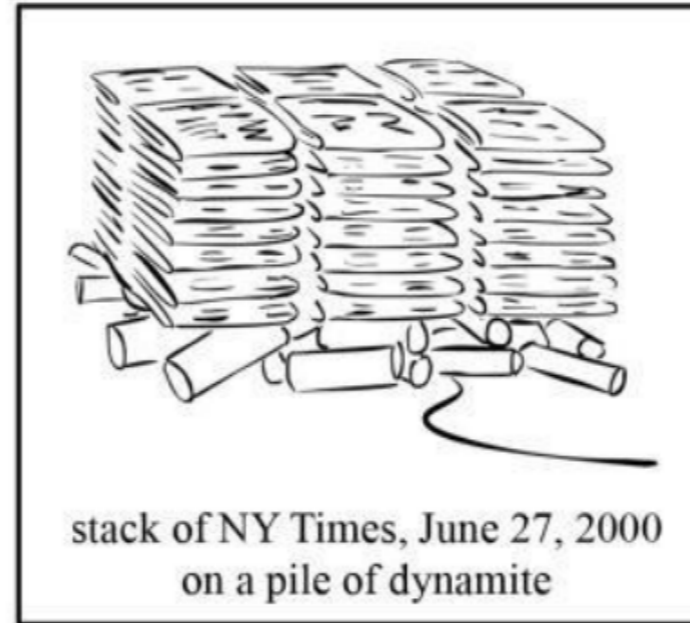
- вирусы
- бактерии
- органеллы

Несколько примеров

- человек (3.2 Гб)<sup>23</sup>
- *E. coli* (4.6 Мб)<sup>1</sup>
- дрозофила (139.5 Мб)<sup>4</sup>
- арабидопсис (135 Мб)<sup>5</sup>
- *Paris japonica* (150 Гб)<sup>40</sup>

(Вороний глаз японский)

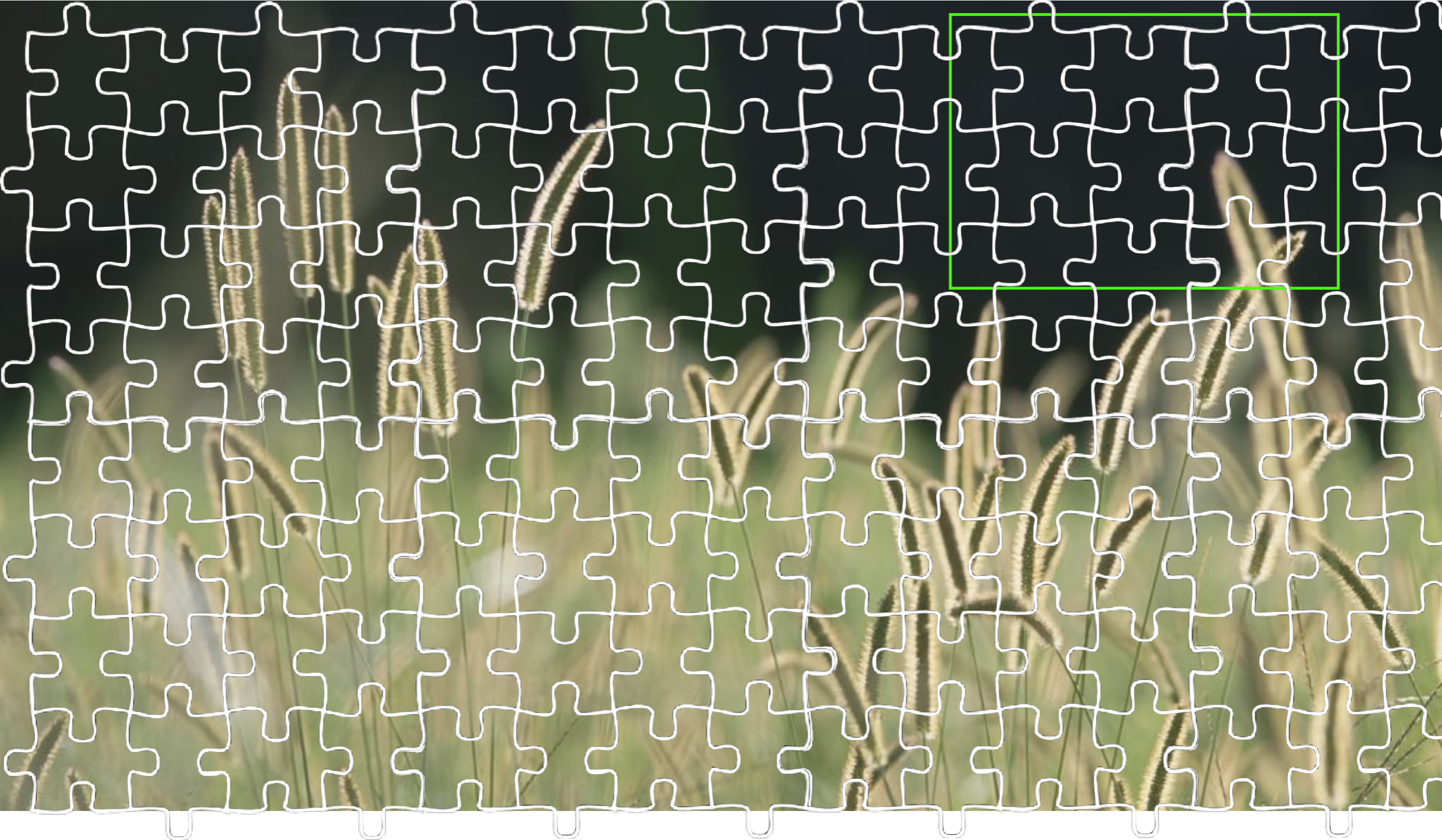
# Задача сборки



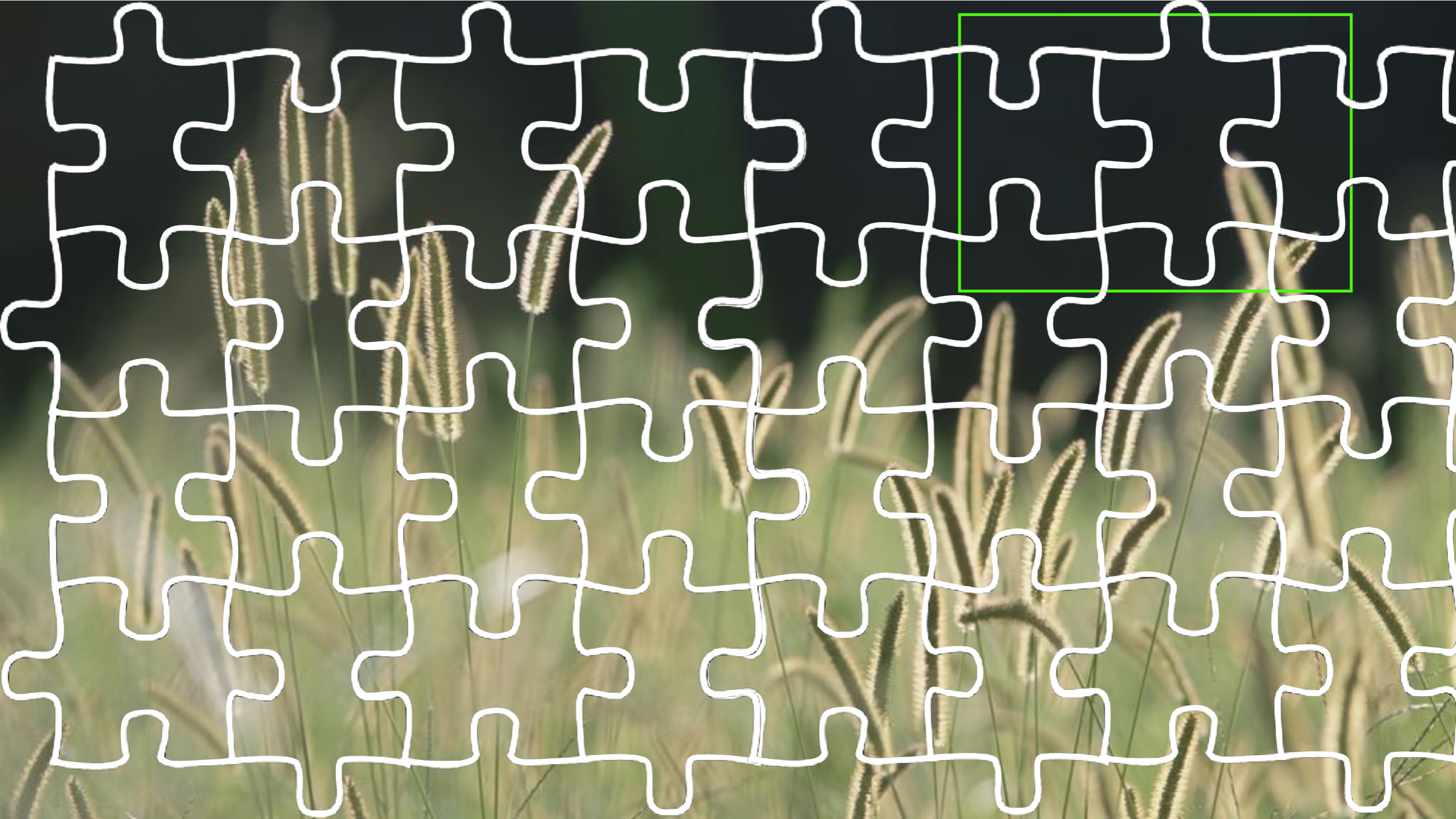
# Размер чтений имеет значение

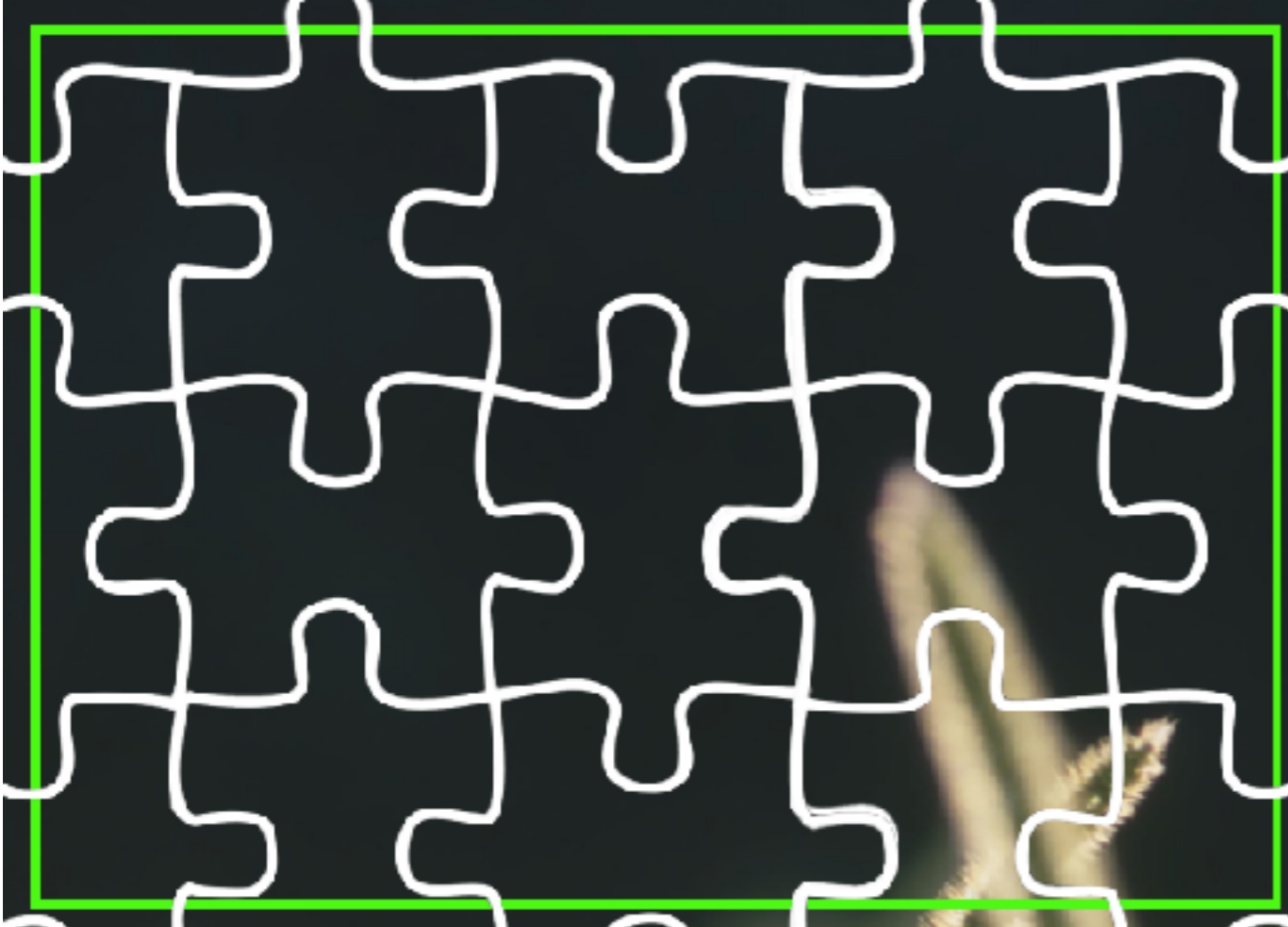


# Размер чтений имеет значение



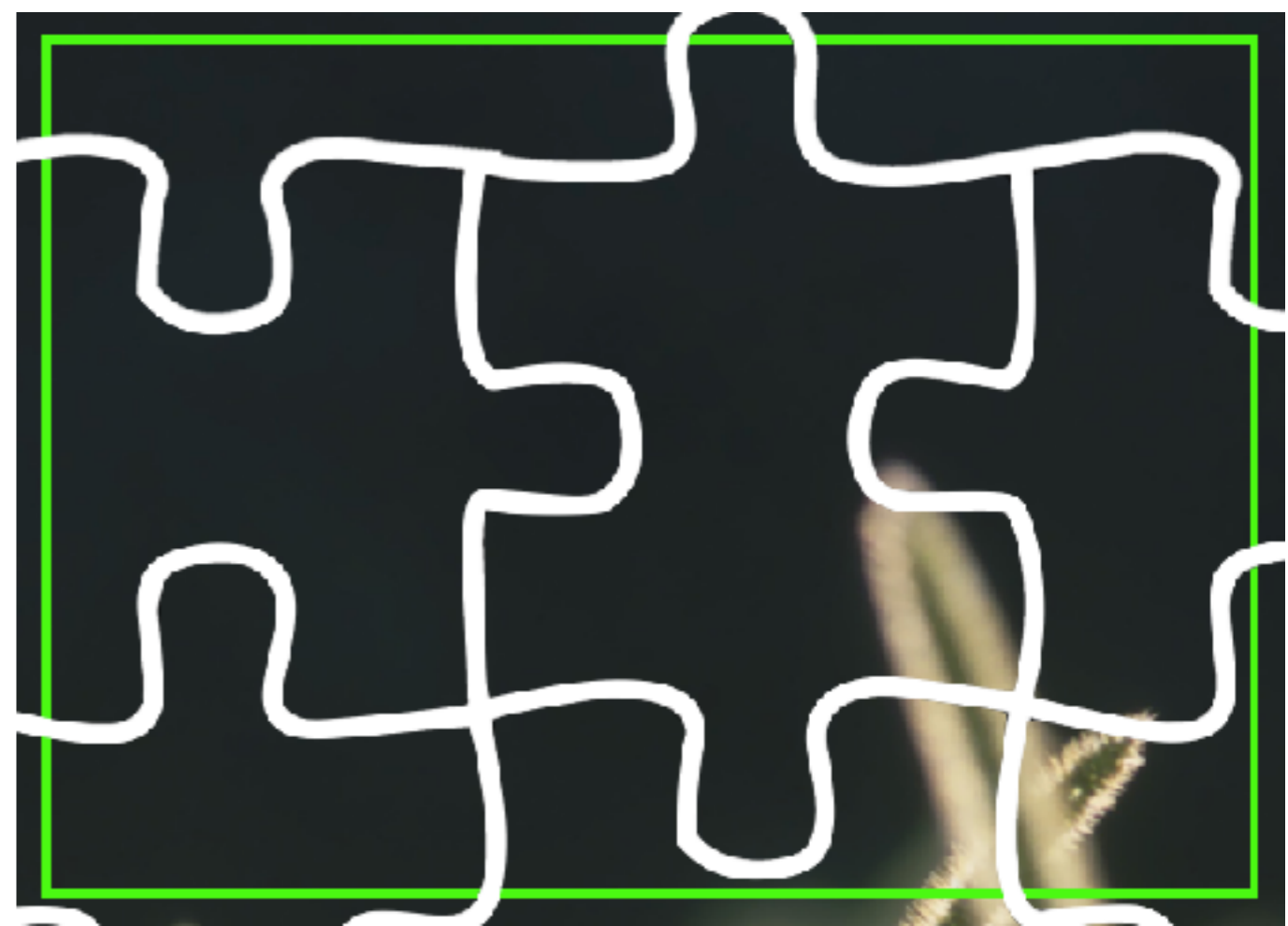
# Размер чтений имеет значение





Почему эта аналогия  
имеет место?

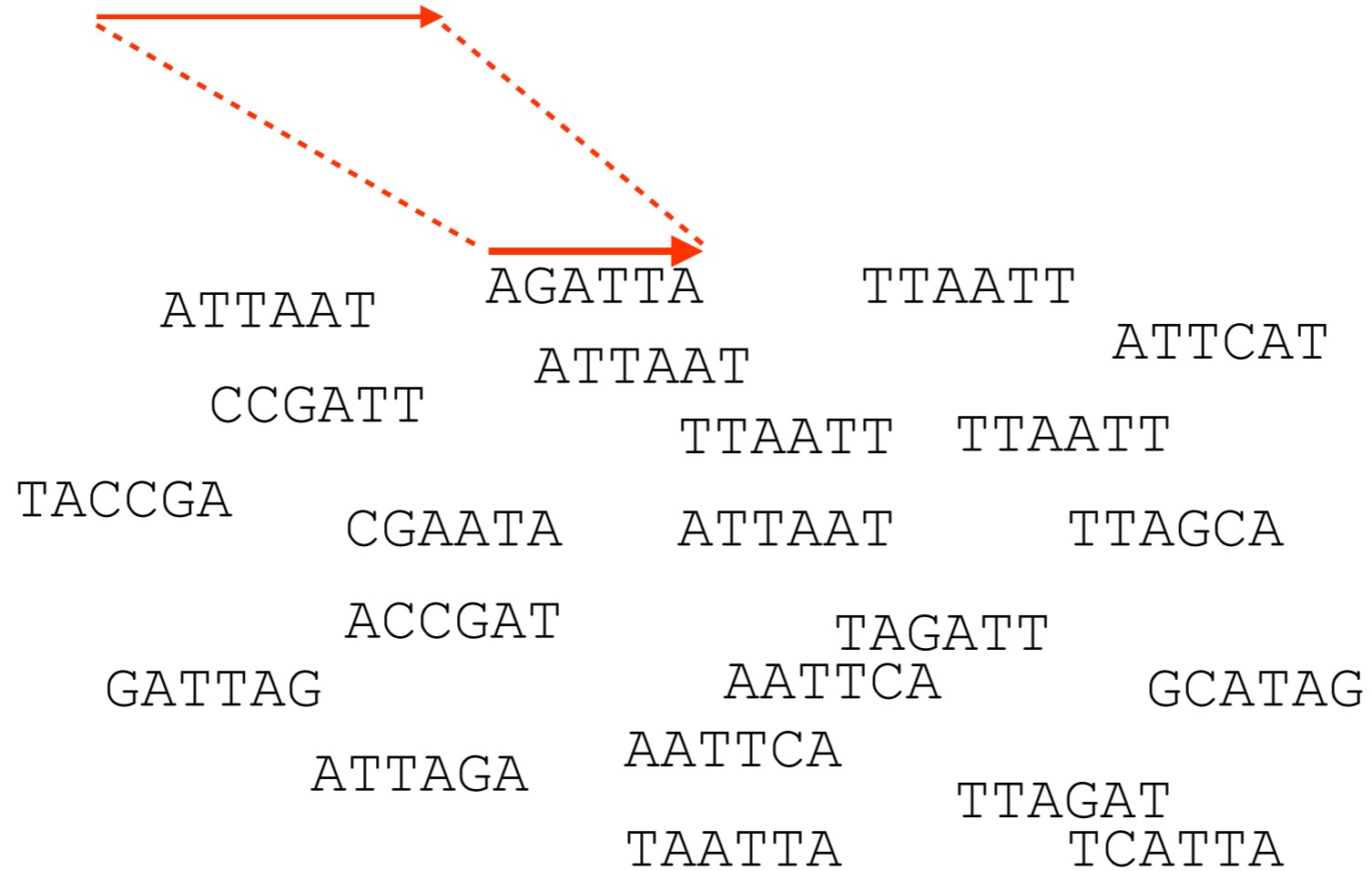
Ответ в алгоритме  
(как всегда ;) )



# Сборка

## *How-to*

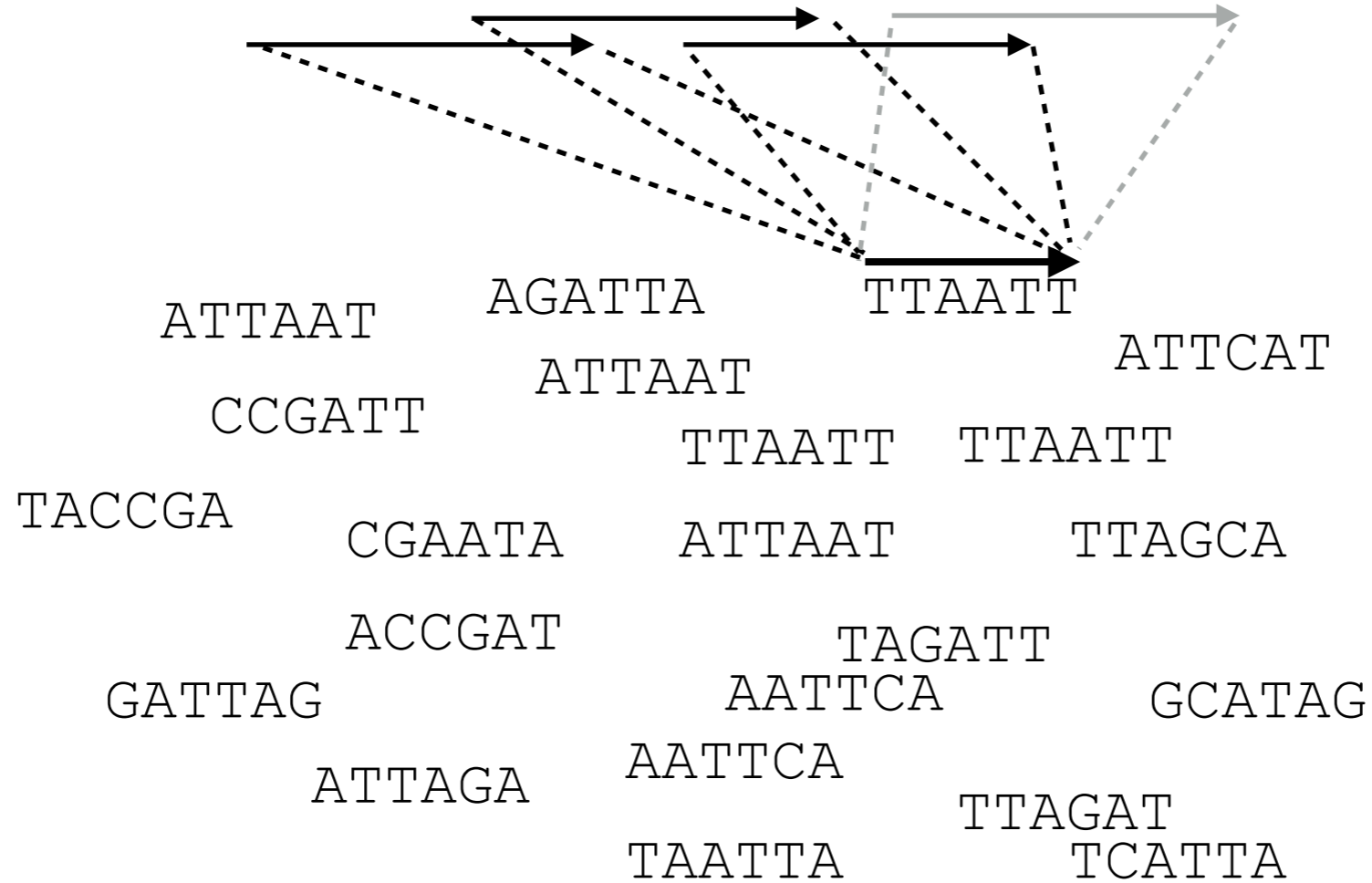
TACCGATTAGATTAAATTAATTAATTCATTAGCATAGCA



# Сборка

## *How-to*

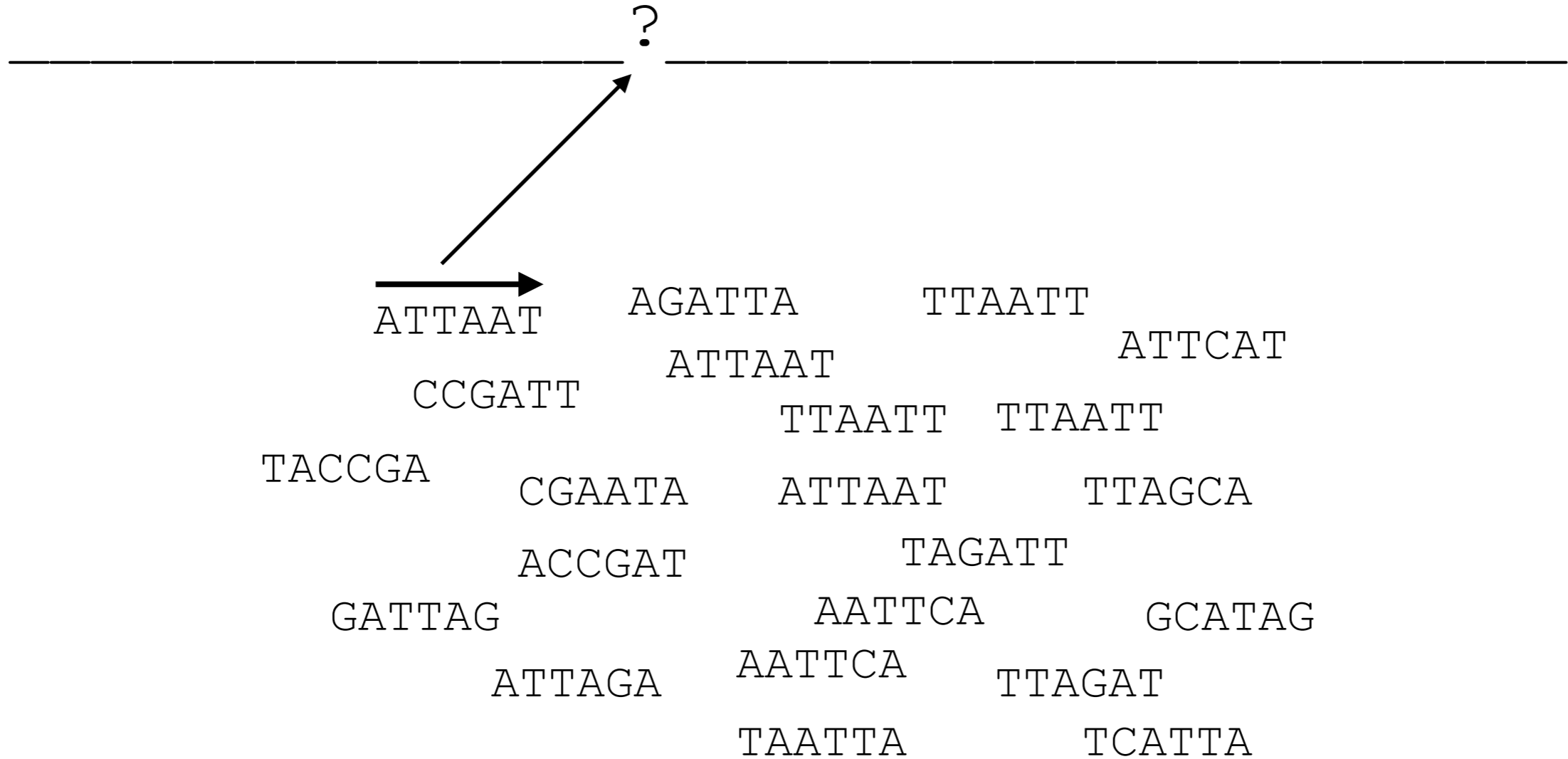
TACCGATTAGATTAAATTAATTAATTCATTAGCATAGCA





# Сборка

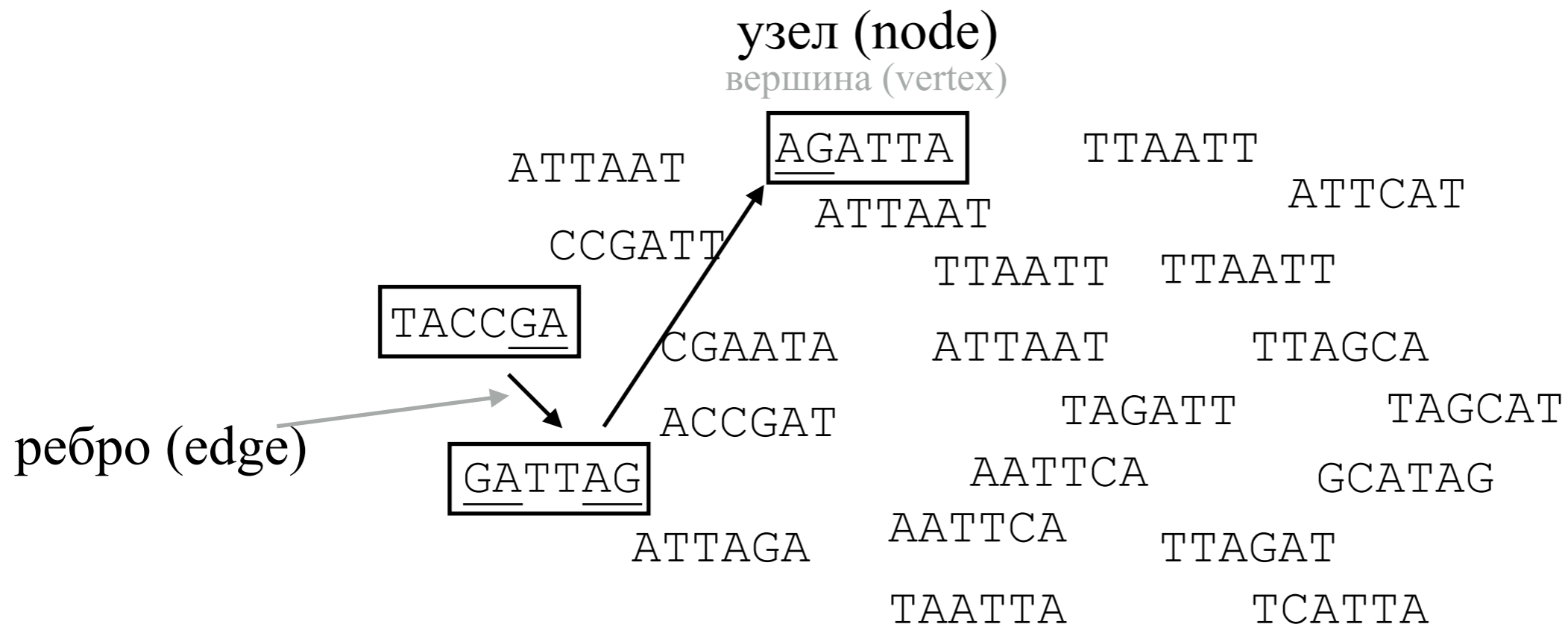
*Что такое геном?*



# Теория графов

*Как собрать пазл?*

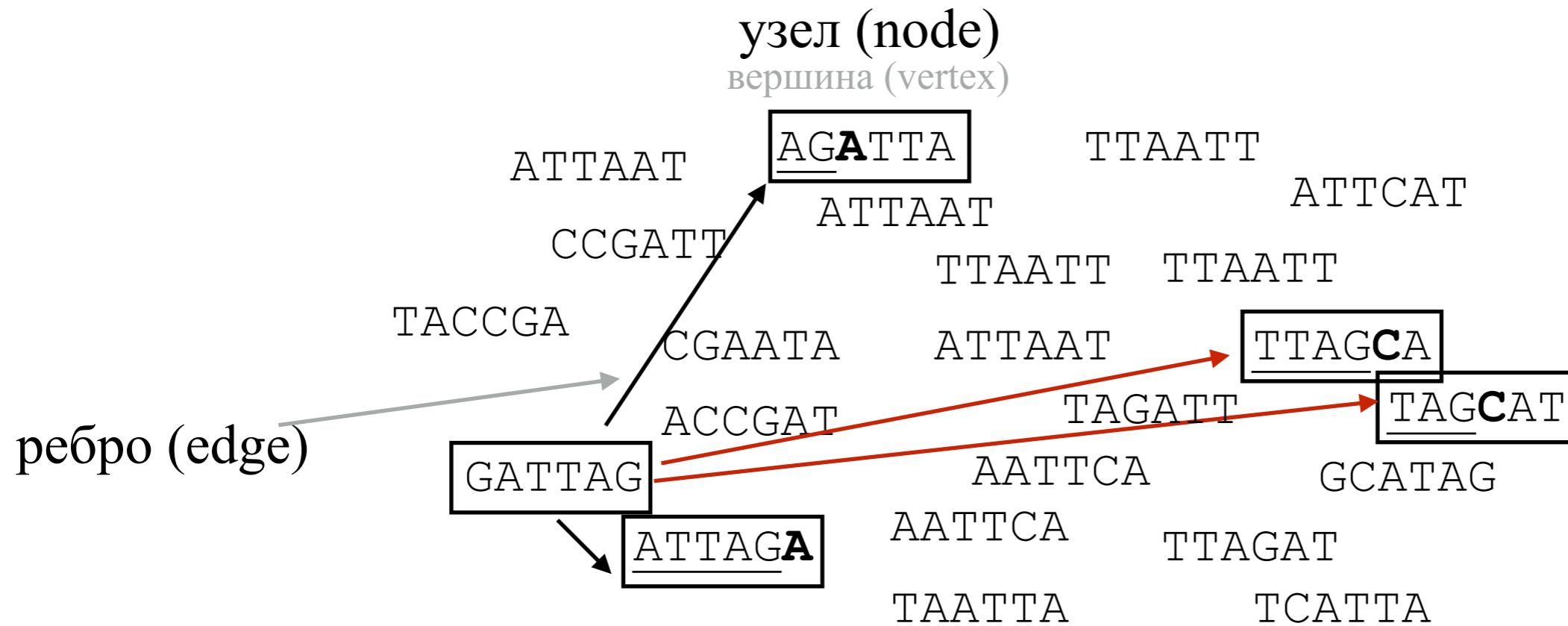
TACCGATTAGATTAAATTAATTAATTCATTAGCATAGCA



# Теория графов

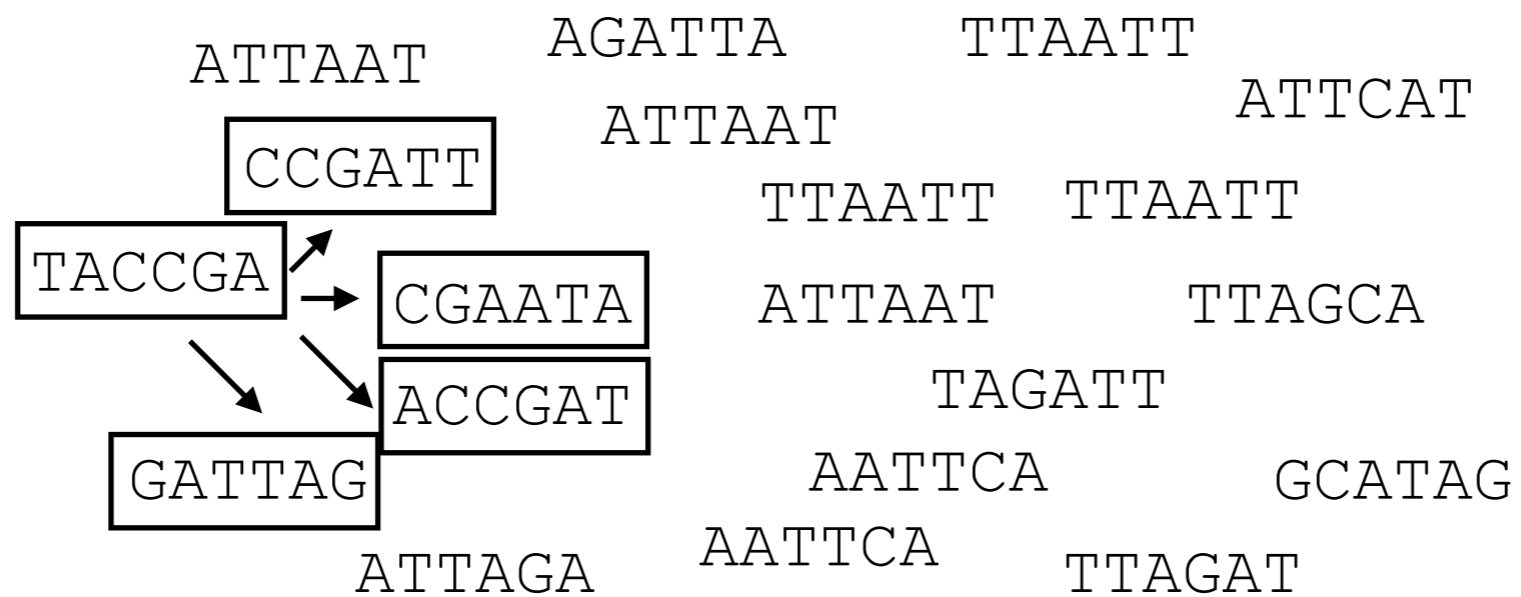
## Первые сложности

TACCGATTAGATTAAATTAATTAATTCATTAGCATAGCA



# Алгоритм OLC

*Overlap - Layout - Consensus*



TACCGA  
 CCGAT**T**  
 CGAATA  
 ACCGAT**T**  
 GAT**T**AG

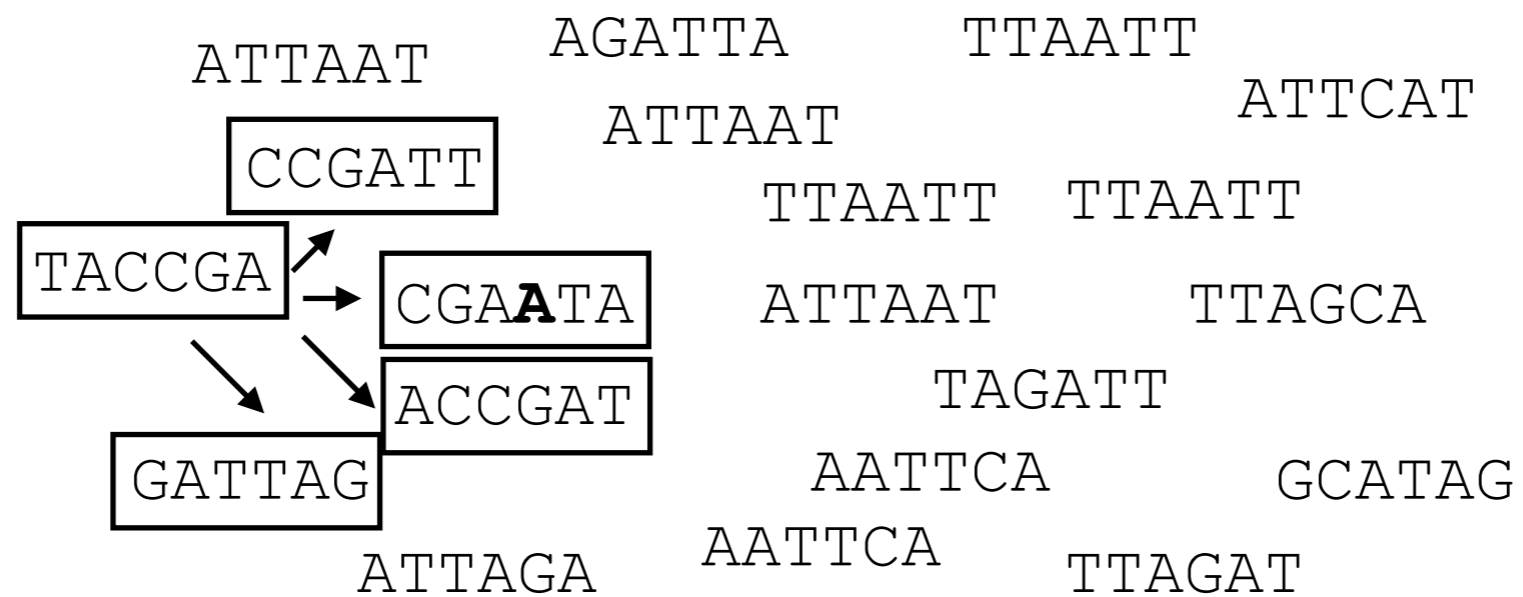
---

TACCGAT**T**AG

# Алгоритм OLC

*Overlap - Layout - Consensus*

**TACCGATTAG**ATTAAATTAATTAATTCATTAGCATAGCA



TACCGA  
 CCGATT  
 CGA**A**TA  
 ACCGAT  
 GATTAG

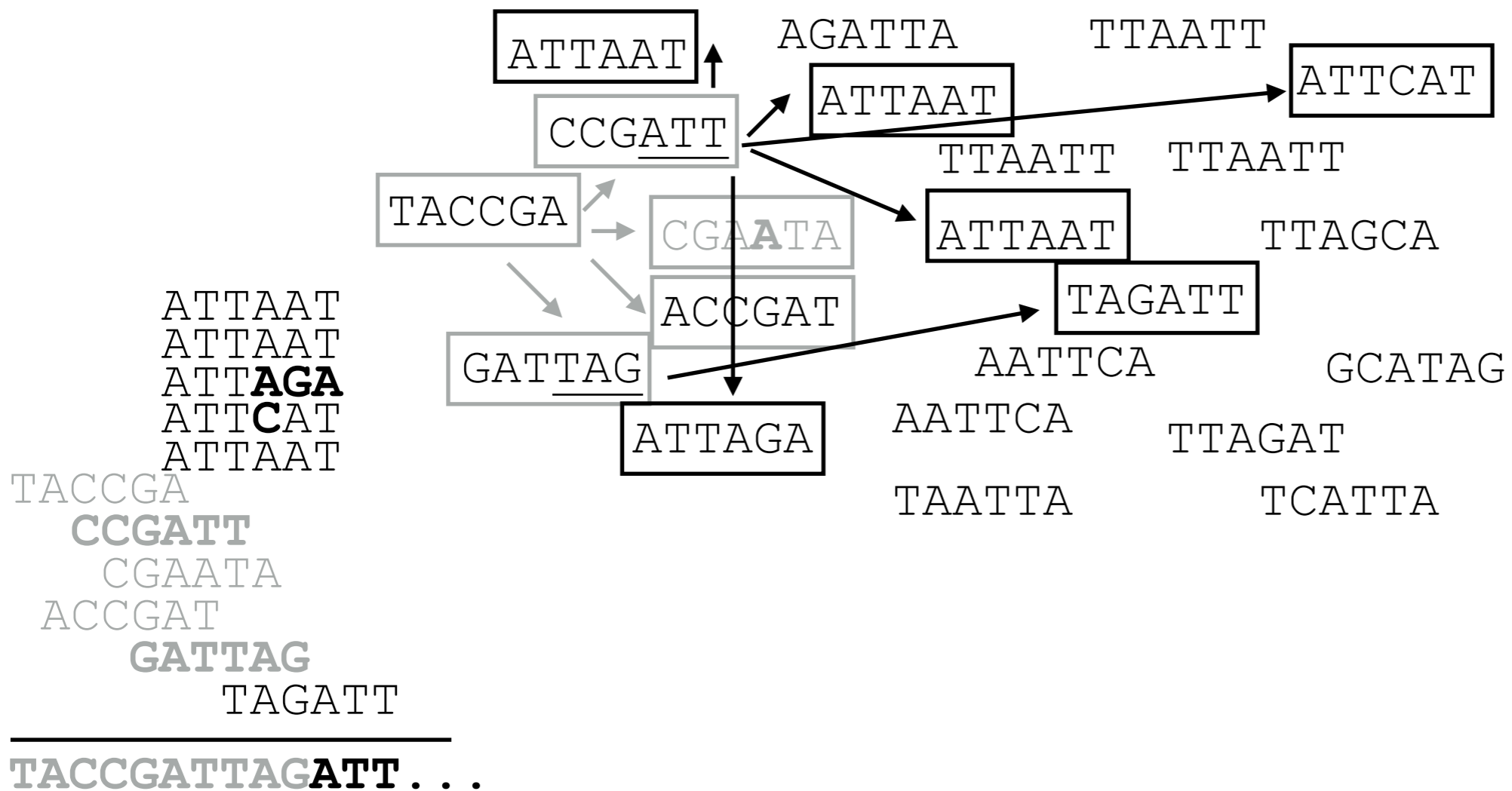
---

TACCGATTAG

# Алгоритм OLC

*Overlap - Layout - Consensus*

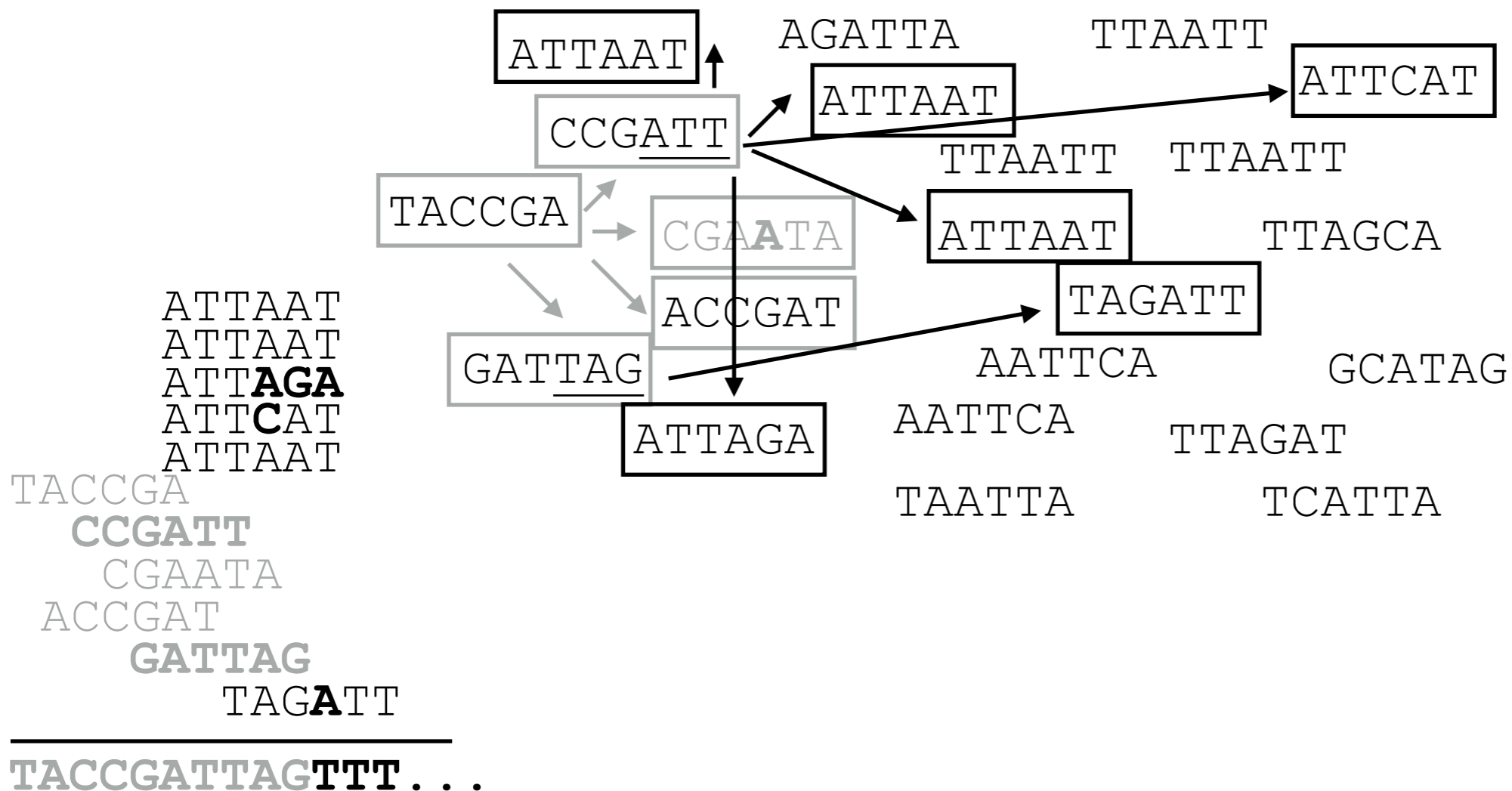
**TACCGATTAG...**



# Алгоритм OLC

*Overlap - Layout - Consensus*

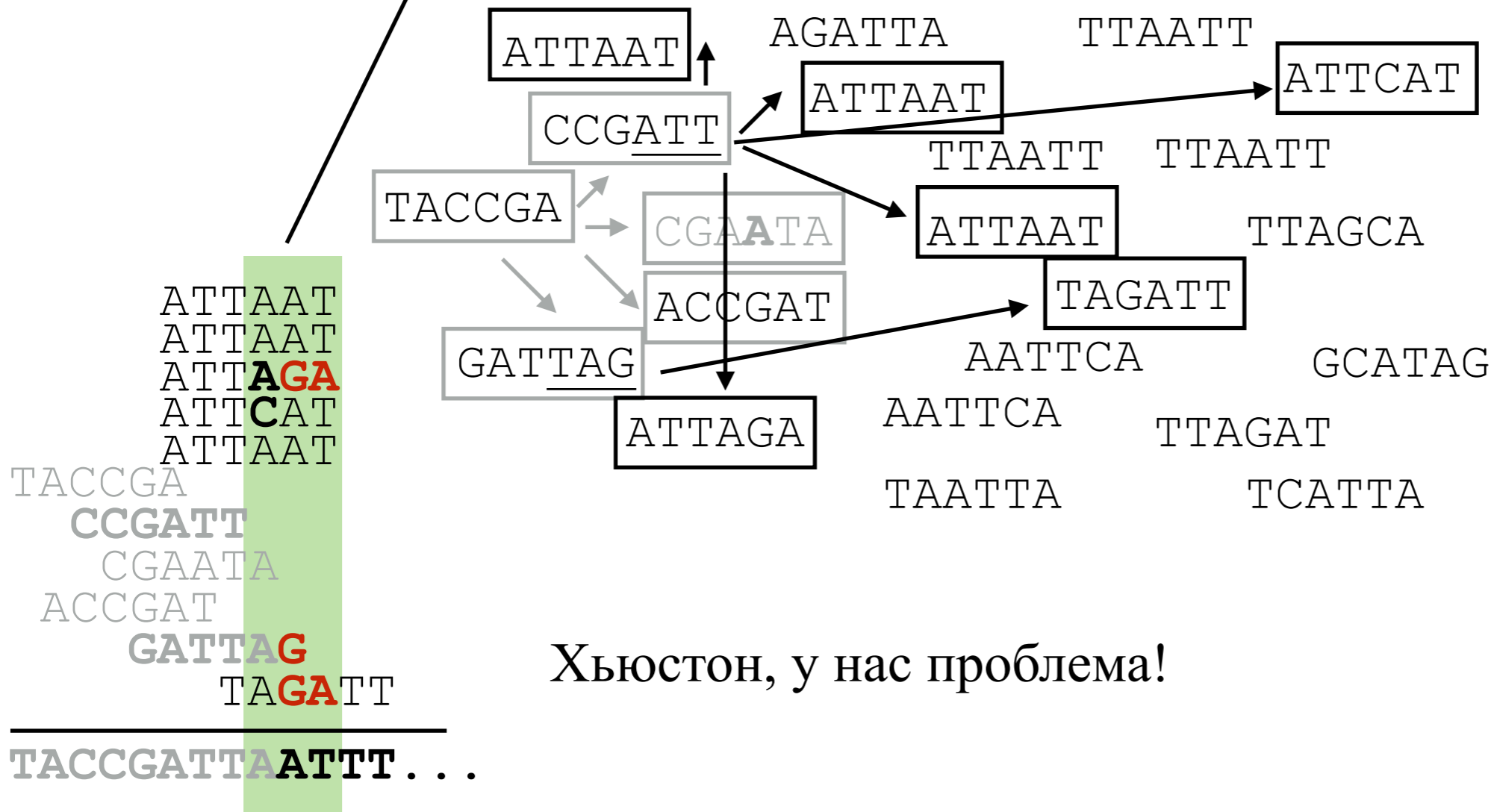
TACCGATTAG**ATT**AATTAATTAATTTCATTAGCATAGCA



# Алгоритм OLC

*Overlap - Layout - Consensus*

TACCGATTAGATT AATTAATTAATTCATTAGCATAGCA

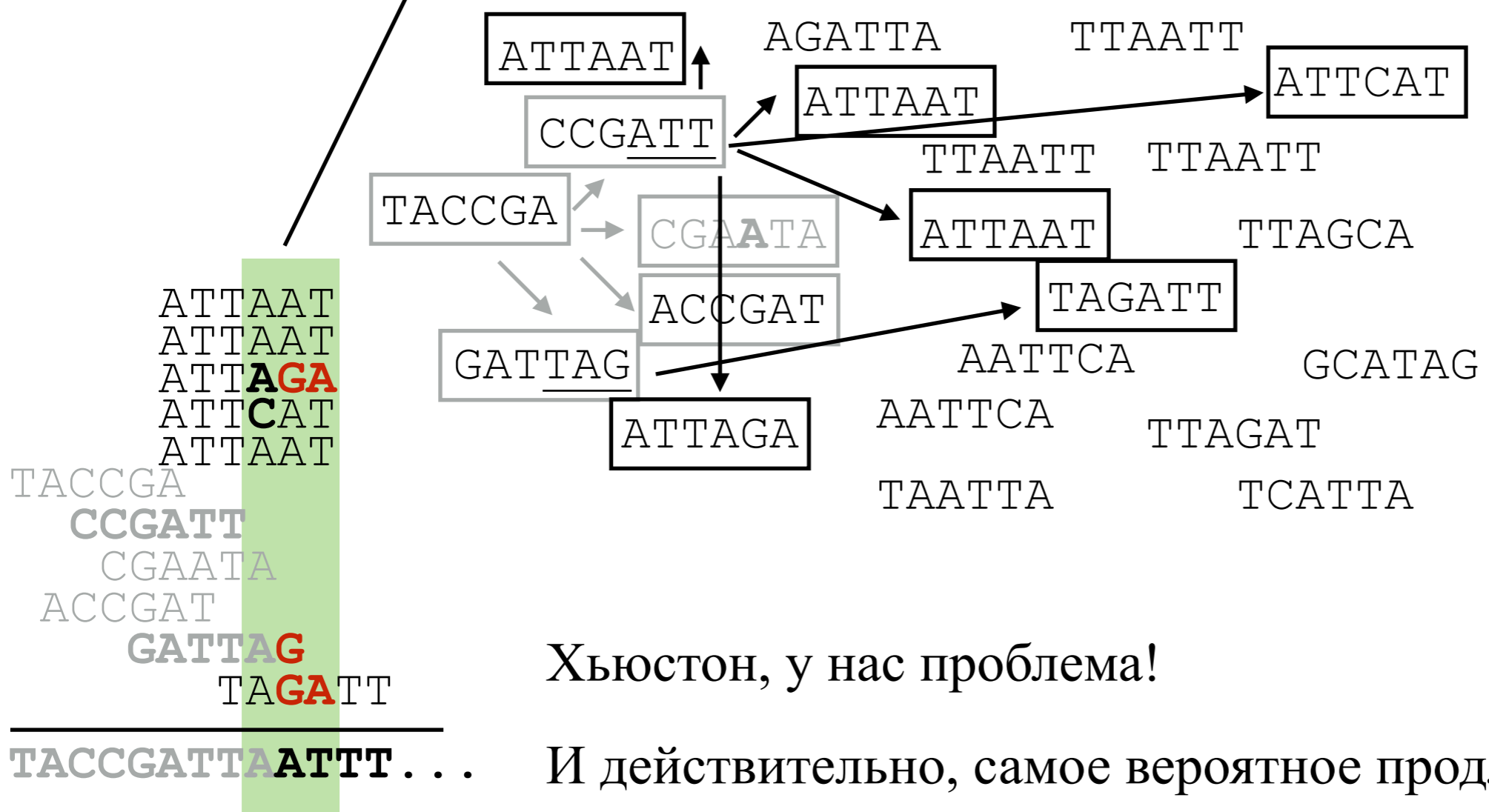




# Алгоритм OLC

*Overlap - Layout - Consensus*

TACCGATTAGATT AATTAATTAATTTCATTAGCATAGCA



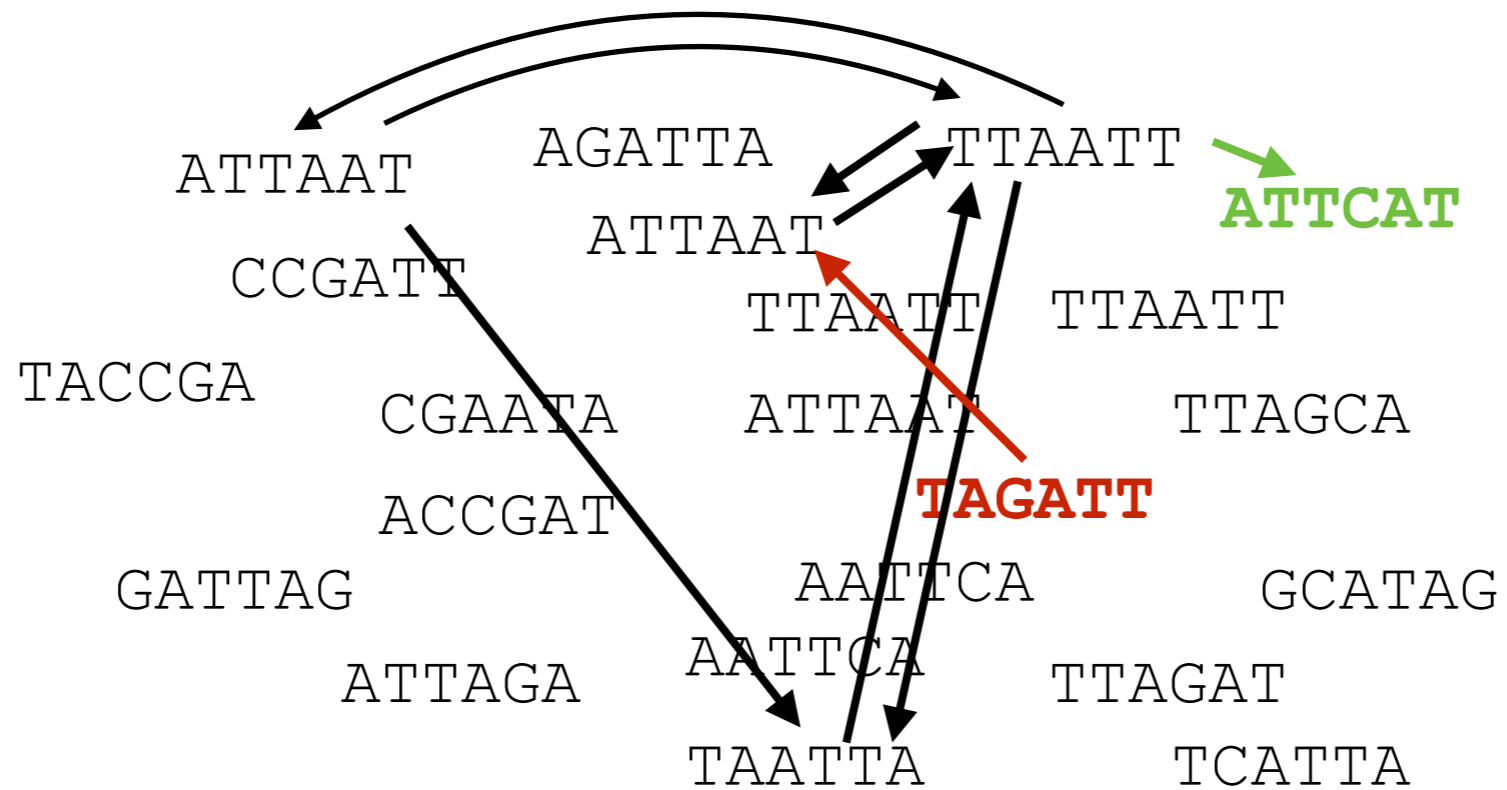
Хьюстон, у нас проблема!

И действительно, самое вероятное продолжение АТТ есть ААТ

# Сборка повторов

*и задача о 7 мостах*

?

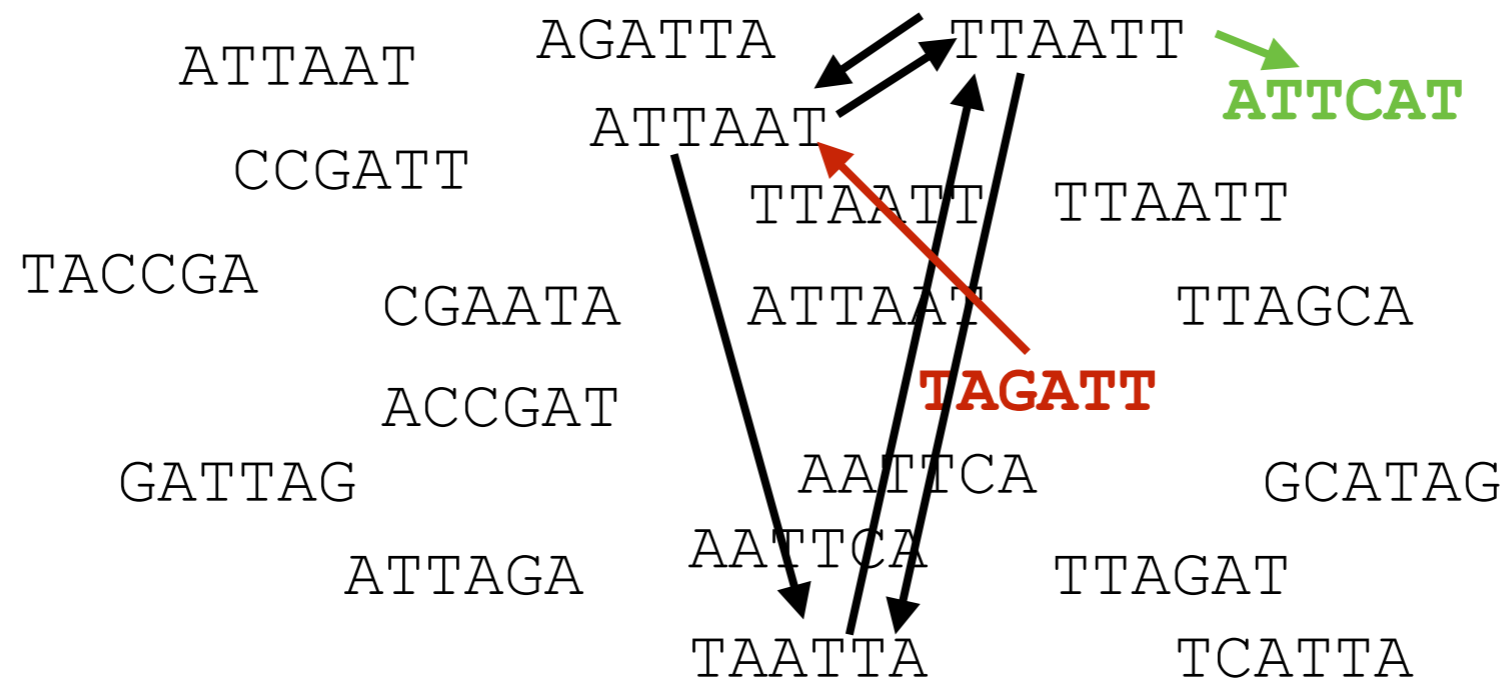


# Сборка повторов

*Как не ошибиться?*

АТТА АТТА АТТА

ТАССГАТ**ТАГАТТ**ААТТААТТА**АТТСАТ**ТАГСАТАГСА

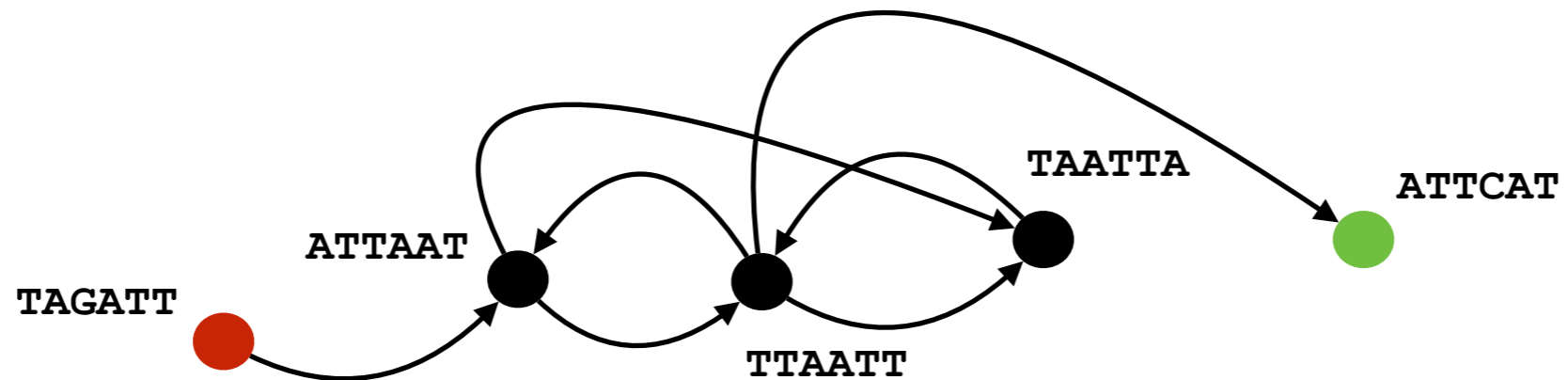


# Сборка повторов

*Как не ошибиться?*

АТТА АТТА АТТА

ТАССГАТ**ТАГАТТ**ААТТААТТА**АТТСАТ**ТАГСАТАГСА

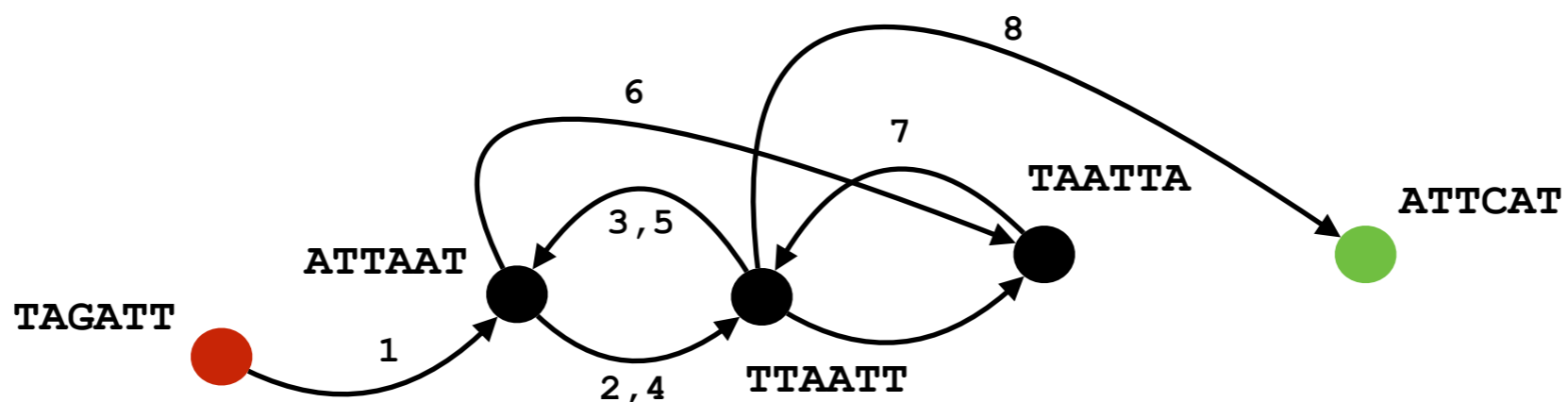


# Сборка повторов

*... ошибка вышла ...*

АТТА АТТА АТТА

ТАССГАТ**ТАГАТТ**ААТТААТТА**АТТКАТ**ТАССГАТАССА

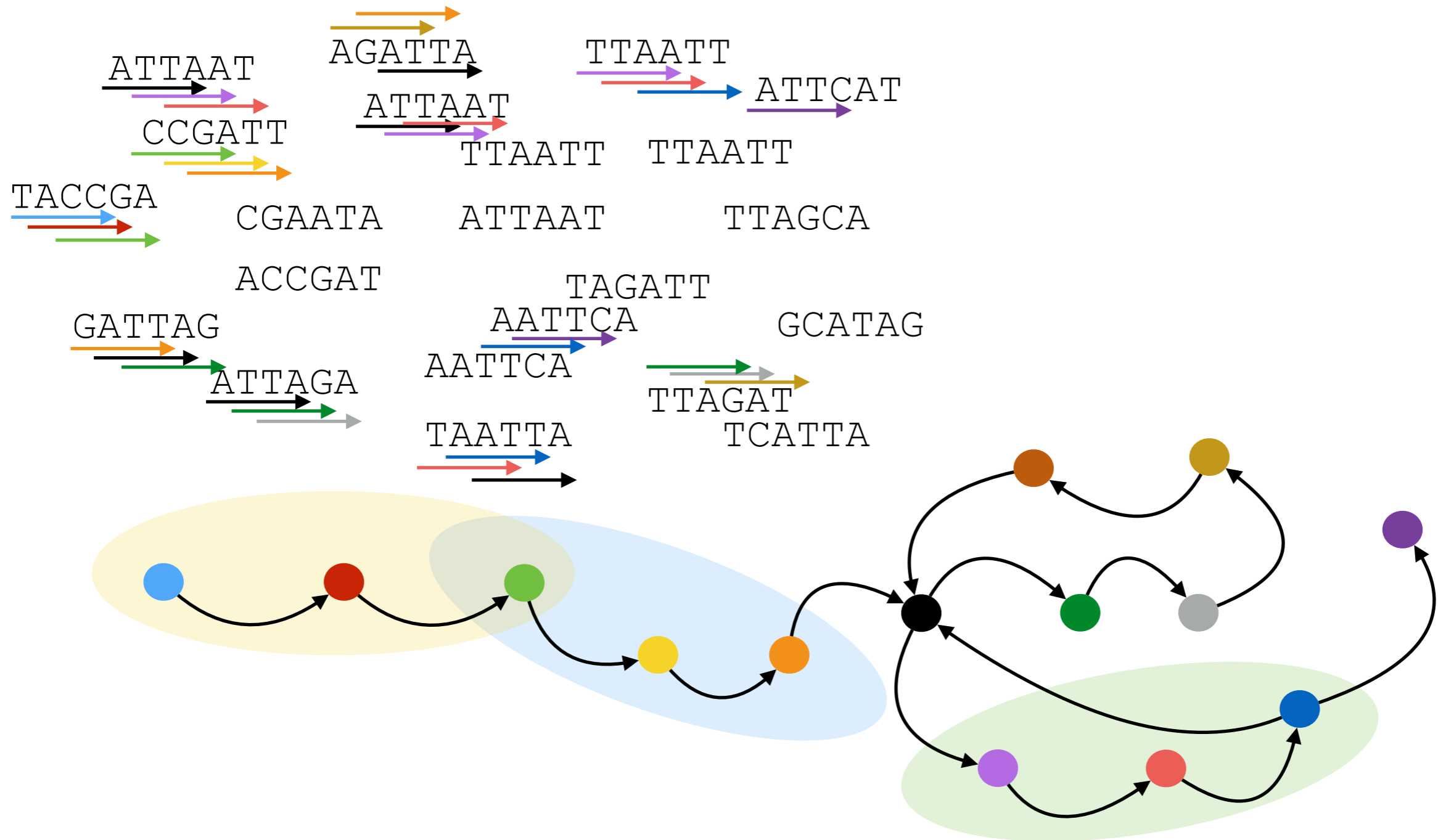


ТАССГАТ +ААТ (1) +Т (1) +ААТ (2) +Т (2) + ААТ (3) + ТА + АТТ (3) +КАТ =  
**ТАГААТ****ААТТААТТА**АТТА**АТТКАТ**

	<b><u>АТТААТ</u></b>	АГАТТА	<b>ТТААТТ</b>	
	ССГАТТ	<b><u>АТТААТ</u></b>		АТТКАТ
ТАССГА		<b>ТТААТТ</b>	<b>ТТААТТ</b>	
	ССГААТА	<b><u>АТТААТ</u></b>		ТТАССА
	АССГАТ		ТАССАТТ	
ГАТТАГ		ААТТССА		ССАТАГ
	АТТАГА	ААТТССА		ТТАССАТ
		ТААТТА		ТССАТТА

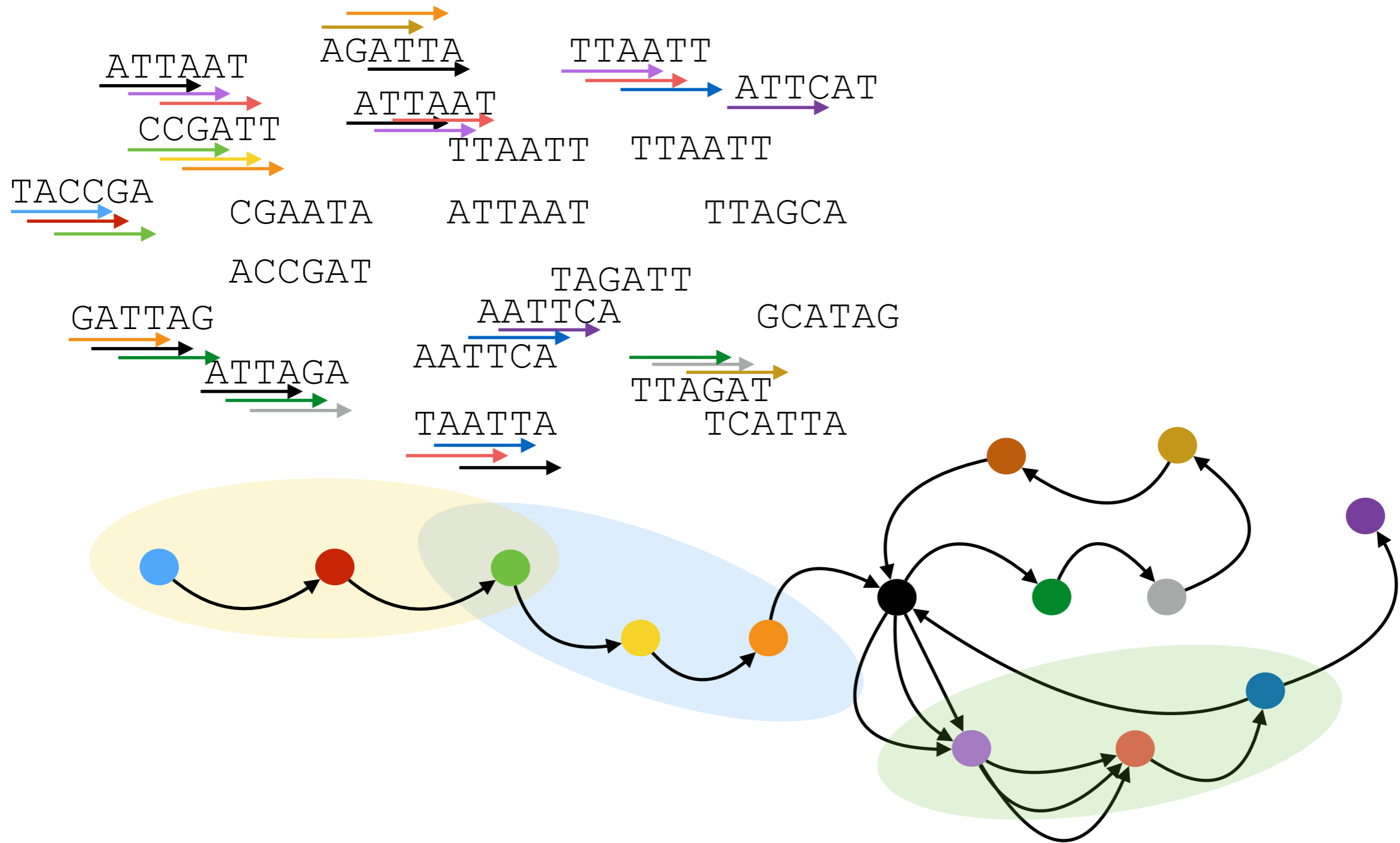
# Сборка методом DBG

## Графы де-Брёйна



# Сборка методом DBG

## Графы де-Брёйна



# DBG или OLC

## *подходы к построению графа*

Что сделать вершиной, а что ребром?

### OLC

Вершины - чтения,  
ребра - перекрытия

(граф быстро растет с числом  
входных данных)

Тип обхода графа:  
обход Гамильтона

(Задача NP-сложная, что  
не позволяет работать на  
больших графах)

Ключевой параметр:  
длина перекрытия

### DBG

Вершины - kмер-ы,  
ребра - чтения

(число вершин известно заранее,  
но оно быстро растет с ростом  $k$ )

Тип обхода графа:  
обход Эйлера

(Можно решить за линейное время)

Ключевой параметр:  
длина  $k$ мер-а



# DBG или OLC

## *подходы к построению графа*

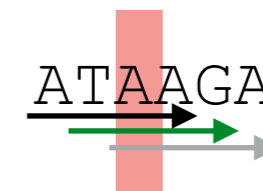
### Сильные и слабые стороны OLC

#### OLC

Большой граф если взять много ридов  
На время построения графа чтений  
негативно влияет число, а не длина ридов  
Путь Гамильтона очень сложно искать  
Довольно неочевидные способы  
упрощения

#### DBG

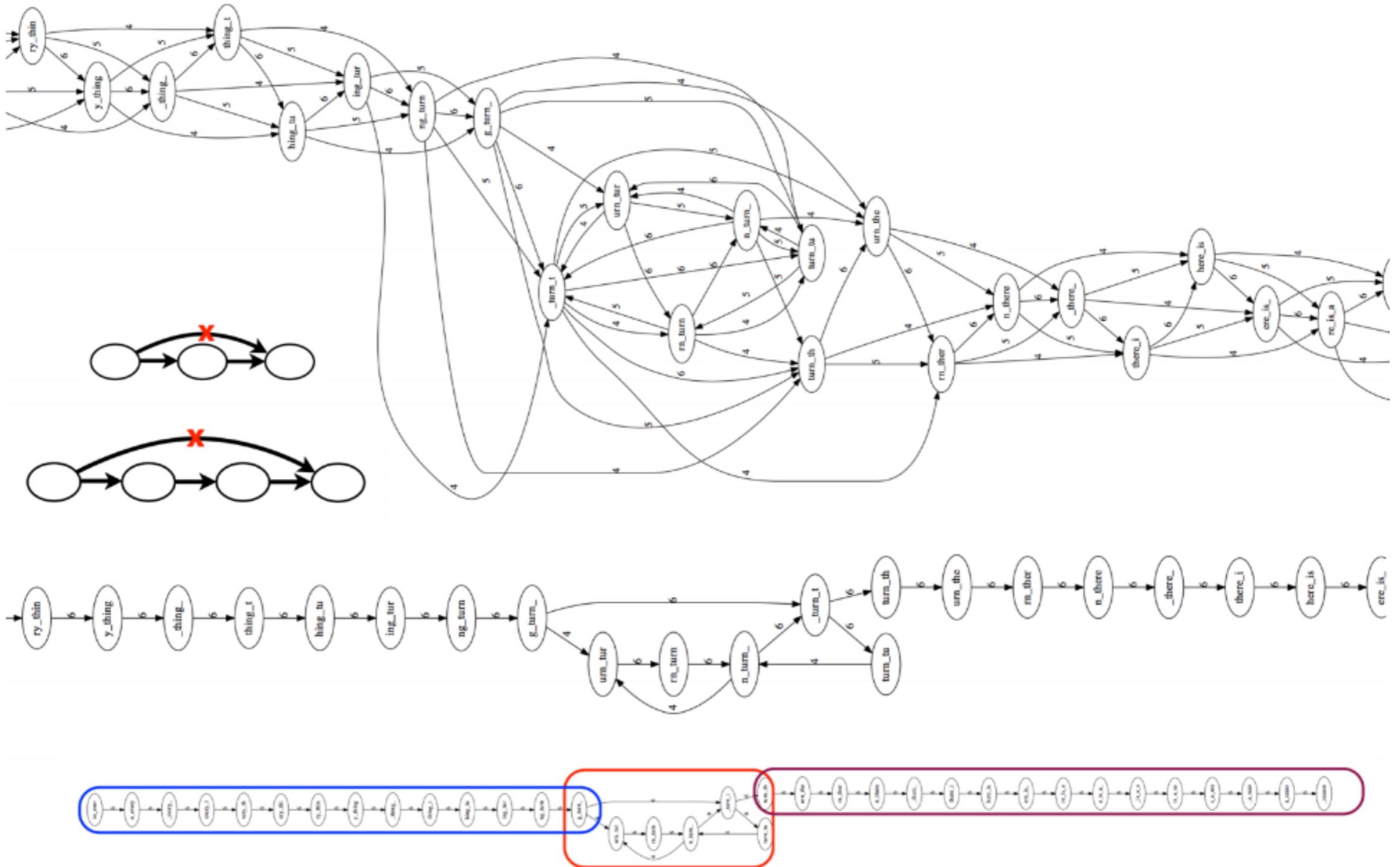
Риды в любом случае делятся на  
короткие фрагменты (kmer)  
Ошибка чтения присутствует сразу  
в нескольких kmer-ах



Относительно легко построить граф

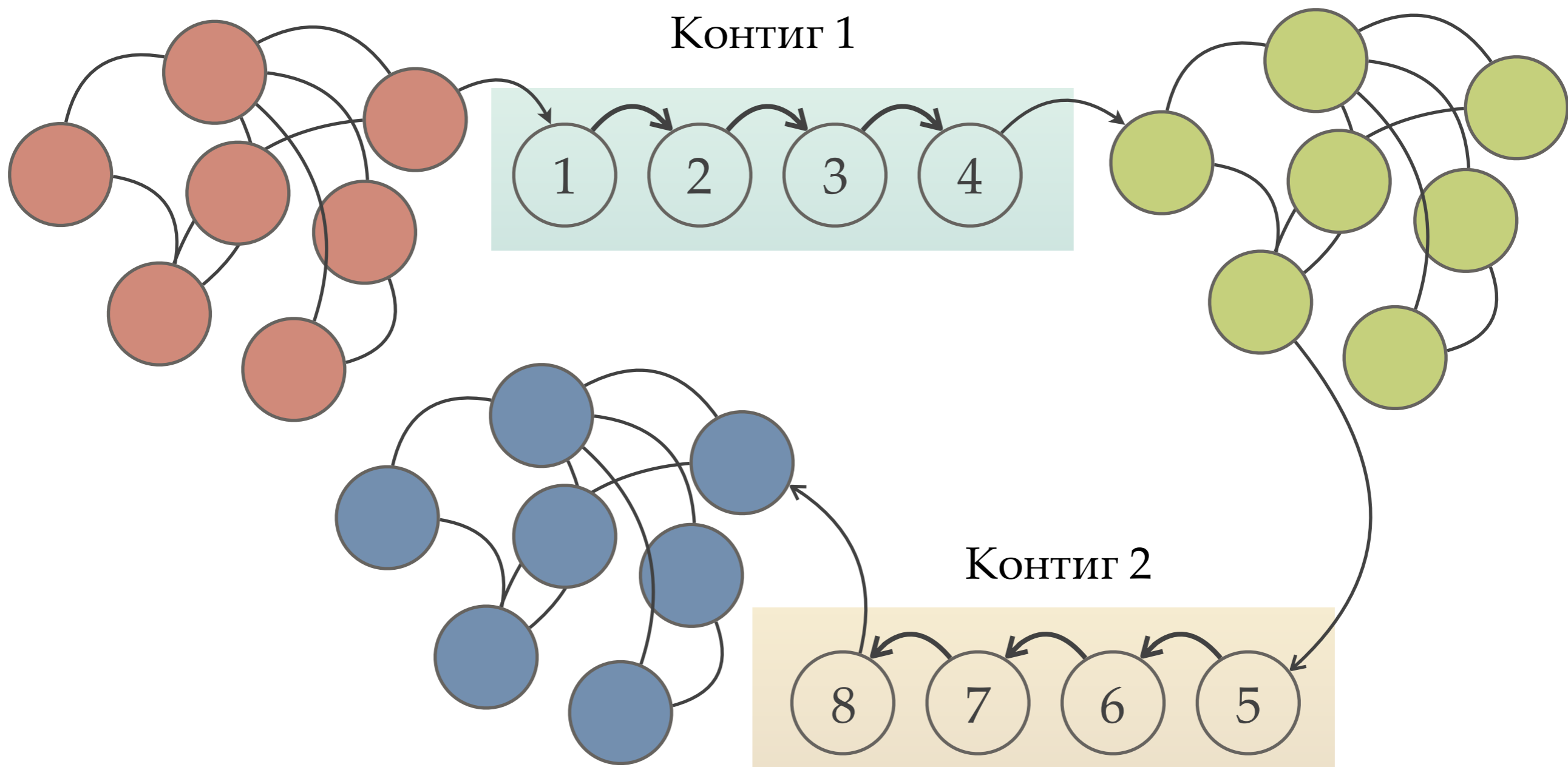
# Упрощение графа

*удаляем всё лишнее*



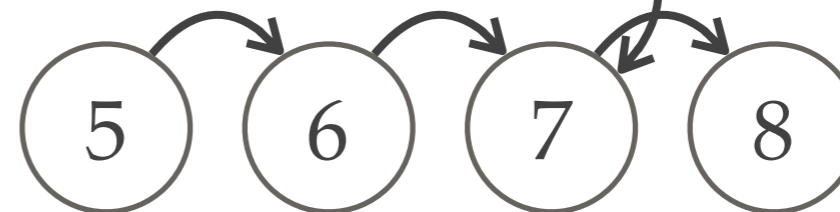
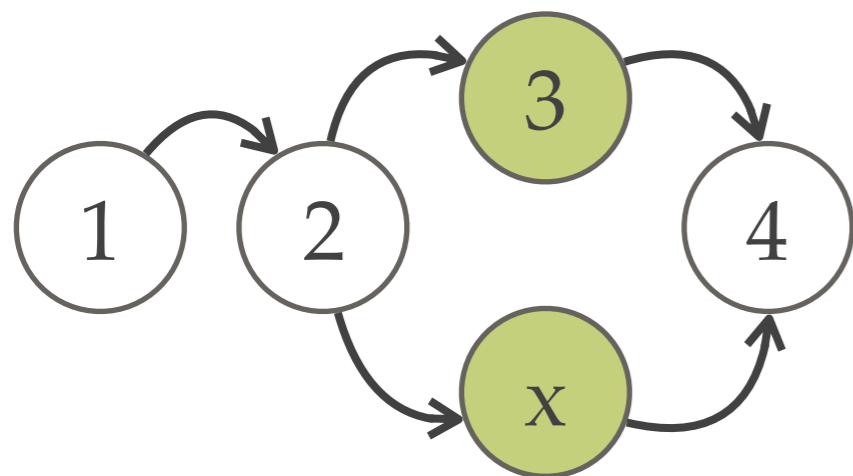
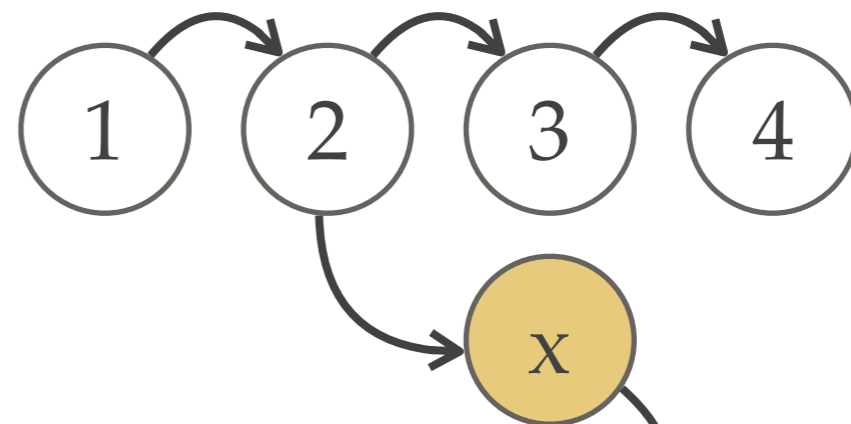
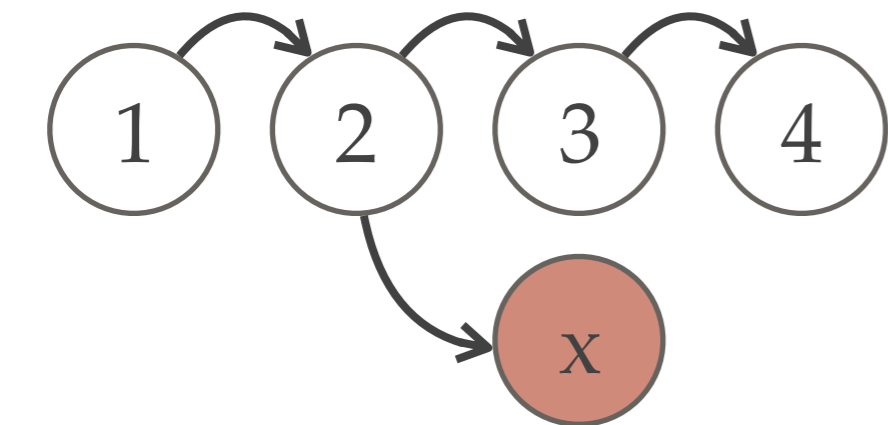
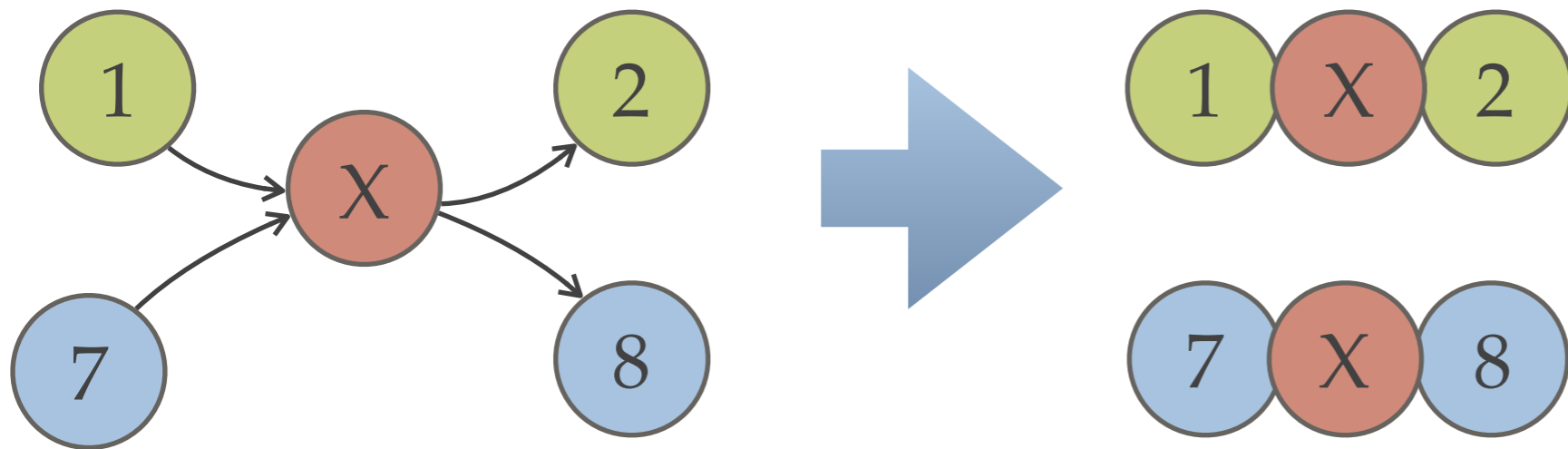
# КОНТИГИ

*когда нет другого пути*



# Упрощение графа

*работа над ошибками*

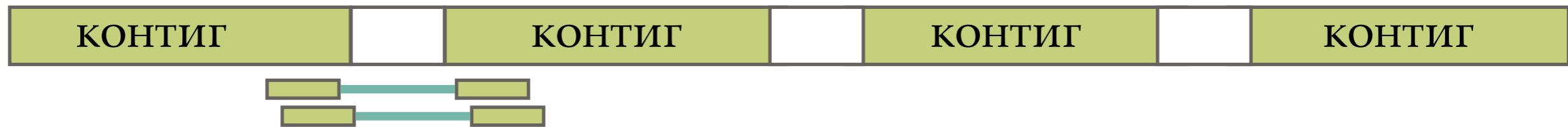


# Скаффолдинг

*От контигов к хромосомам*

Задача процесса скаффолдинга - создание макета хромосомы (в идеале)

*скаффолд*



Сборщик может не осуществлять скаффолдинг

Скаффолдинг почти всегда имеет дело с парными чтениями

# Качество сборки

*когда нет другого пути*

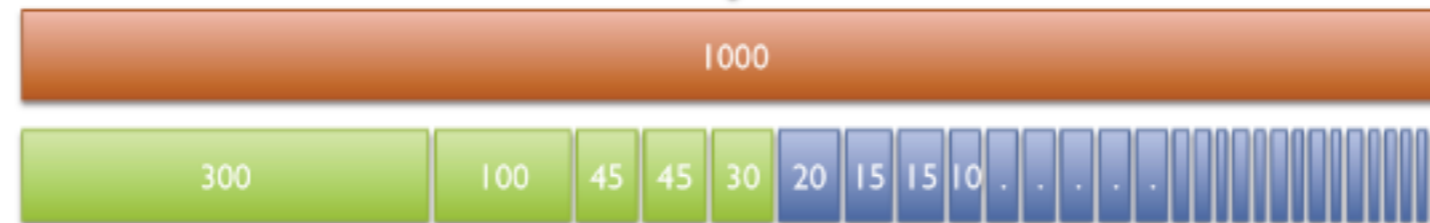
## Статистика N50

Example: 1 Mbp genome

50%

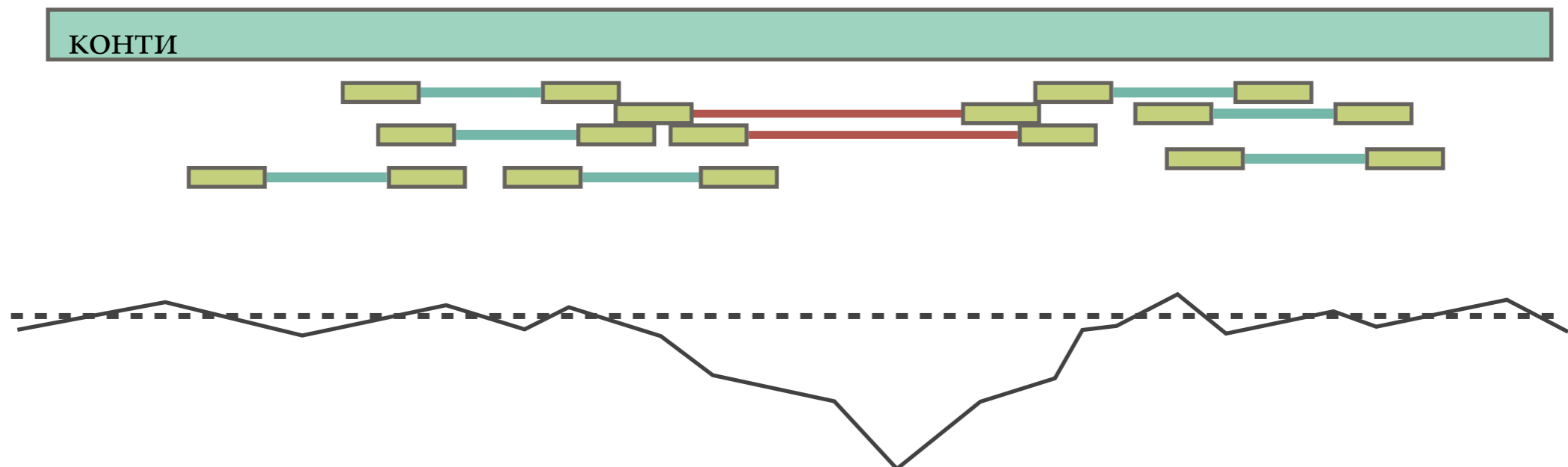


1000



N50 size = 30 kbp

## Картирование чтений



# Улучшение сборки

## *Assembly finishing*

### Оценка качества:

*Картирование чтений, статистика длин элементов, ML-модели, полнота сборки, kmer-ный анализ*

### Автоматическое улучшение:

*Основано на принципе локальной пересборки (gap closing)*

На этом этапе часто используют дополнительные данные (другие библиотеки, парные чтения, карты маркеров)

Существует масса ПО для автоматической коррекции ошибок и закрытия пробелов в сборке

IMAGE, PAGIT, GapCloser, GapFiller, CloG, Sealer

### Курируемая человеком сборка:

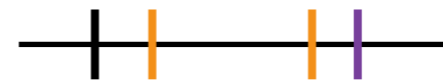
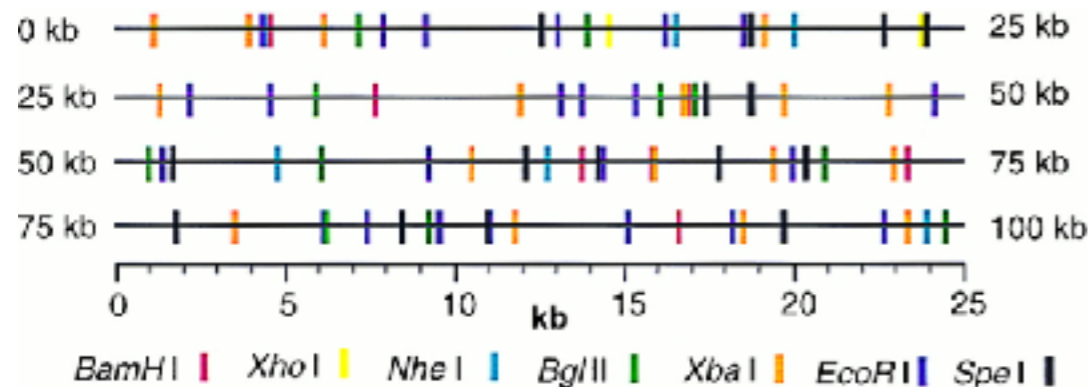
Редактирование сборки оператором на основании имеющихся сведений о поведении неких молекулярных маркеров

Редактирование графа контигов или (реже) графа чтений

# Другие методы сборки

*точнее, скаффолдинга*

- Оптическое картирование (optical mapping)
- Генетические карты
- Карты контактов HiC
- По референсу
- *Транскриптомная сборка*





# Программы-сборщики

*Каждый биоинформатик хотя бы раз в жизни написал свой картировщик или свой сборщик*

Название	Алгоритм	Технологии	Авторы	Представлен
ABYSS	De Bruijn	Solexa, SOLiD	Simpson, J. et al.	2008
ALLPATHS-LG	De Bruijn	Solexa, SOLiD	Gnerre, S. et al.	2011
Celera WGA Assembler / CABOG	OLC	Sanger, 454, Illumina	Myers, G. et al.; Miller G. et al.	2004
CLC Genomics Workbench	String Graph	Sanger, 454, Solexa, SOLiD	CLC bio	2008
Edena	OLC	Illumina	D. Hernandez et al.	2008
Euler	De Bruijn	Sanger, 454 (,Solexa ?)	Pevzner, P. et al.	2001
Euler-sr	De Bruijn	454, Solexa	Chaisson, MJ. et al.	2008
IDBA	De Bruijn	Sanger,454,Solexa	Yu Peng, Henry C. M. Leung, Siu-Ming Yiu, Francis Y. L. Chin	2010
MIRA	OLC	Sanger, 454, Solexa	Chevreur, B.	1998
Newbler	String Graph	454, Sanger, Solexa, Ion	454/Roche	2009
PCAP	OLC	Sanger, 454	Huang et al.	2003
SGA	String Graph	Illumina, Ion Torrent	Simpson, J. et al.	2011
SOPRA		Illumina, SOLiD, Sanger, 454	Dayarian, A. et al.	2010
SOAPdenovo	De Bruijn	Solexa	Li, R. et al.	2009
SPAdes	De Bruijn	Illumina, Solexa	Bankevich, A et al.	2012
Velvet	De Bruijn	Sanger, 454, Solexa, SOLiD	Zerbino, D. et al.	2007