

# **Коррекция ошибок в ридах**

## **Алгоритмы в биоинформатике**

**Антон Елисеев**  
**[eliseevantoncoon@gmail.com](mailto:eliseevantoncoon@gmail.com)**

# Что было в прошлом модуле

- Сравнение двух последовательностей: расстояния, глобальное и локальное выравнивания, штрафы за гэпы, эффективное использование памяти
- Поиск специфичных участков генома при помощи НММ
- Сравнение многих последовательностей между собой и одной последовательности со многими
- Выравнивание на референсный геном (BWT, BWA)
- Перестройки в геноме, синтные блоки
- Введение в филогению

# Что будет в этой модуле

- Секвенирование! NGS данные. Артефакты и важная информация.
- Сборка генома из коротких прочтений.
- Сборка многих геномов. Метагеном, гаплотипы и связанные задачи.
- Эволюция и ее параметры. Зачем нужны вероятностные модели?
- Вторичная структура РНК

# В этой лекции

- Методы секвенирования
- Секвенирование как случайный процесс
- Ошибки секвенирования
- Способы отбрасывать риды с ошибками
- Способы коррекции ошибок

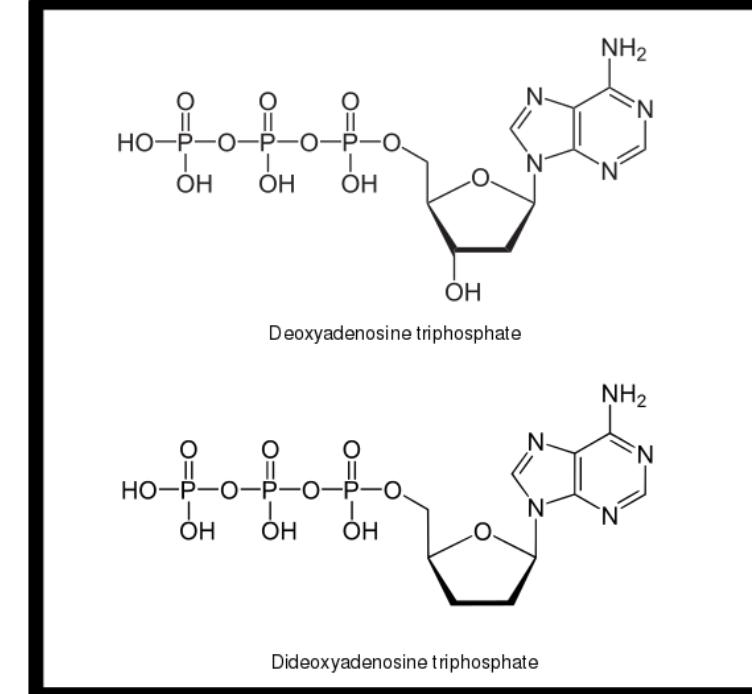
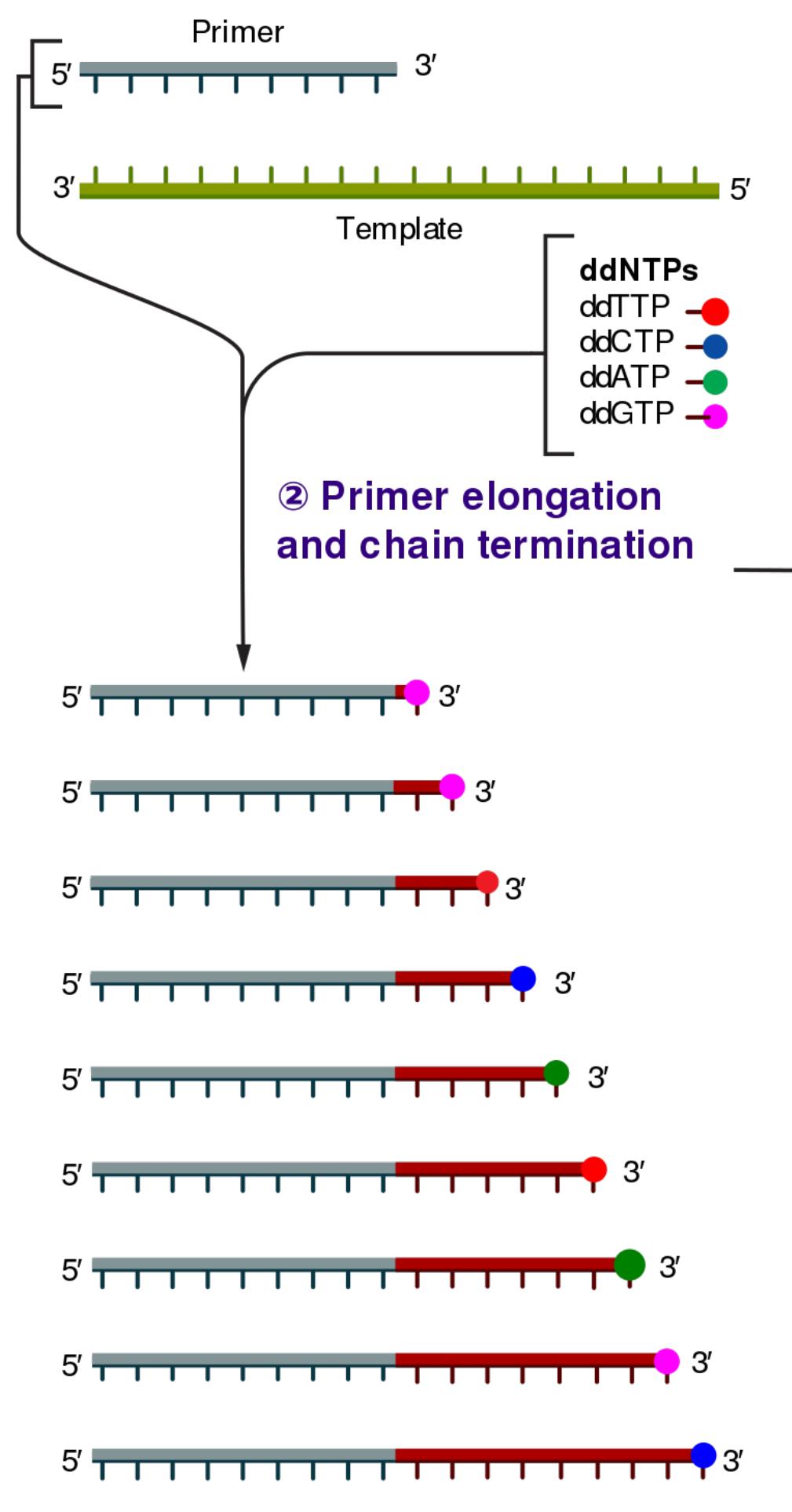
# **Методы**

# Методы

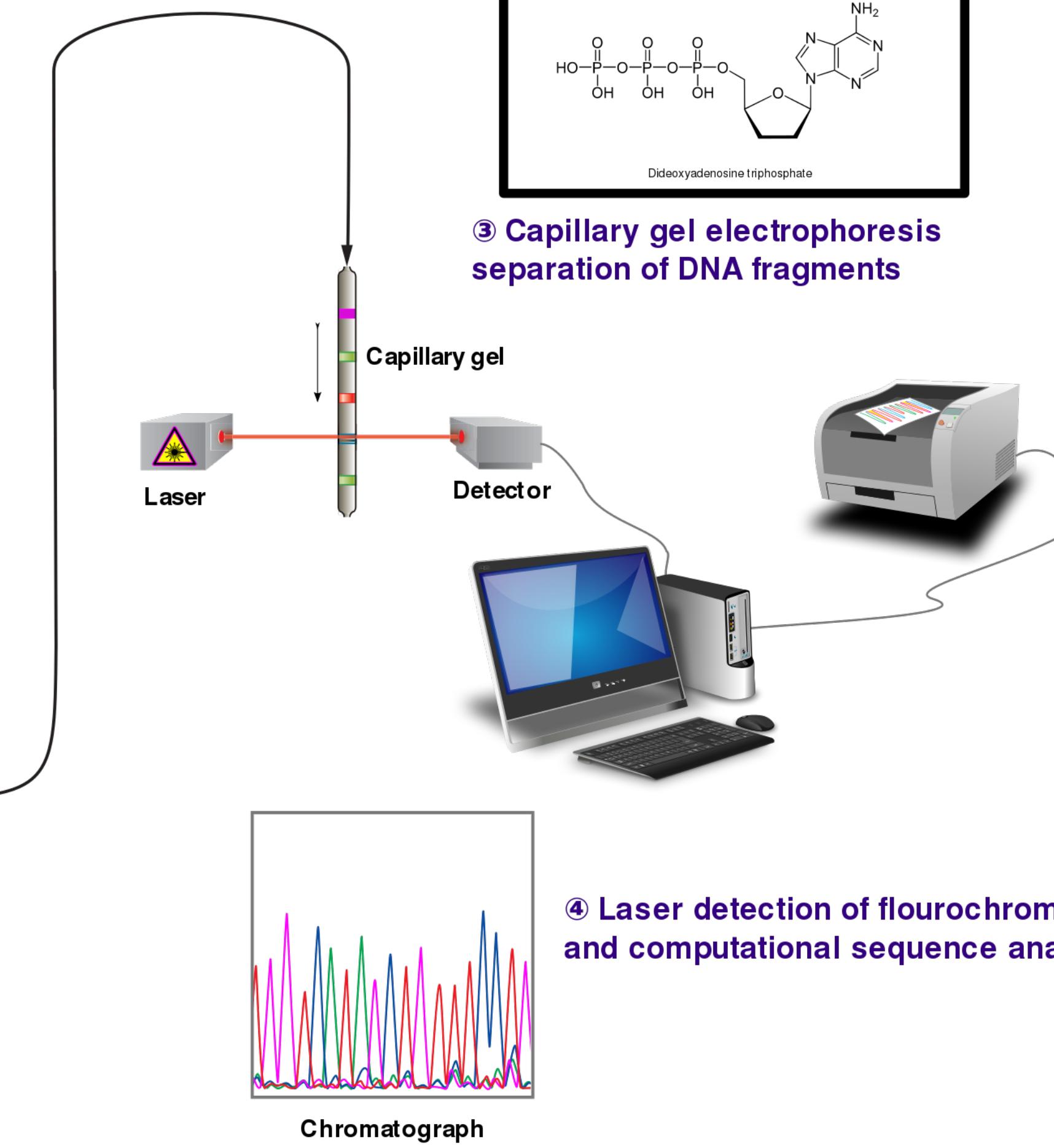
- Sanger
- Illumina
- Nanopore

# Методы: Sanger

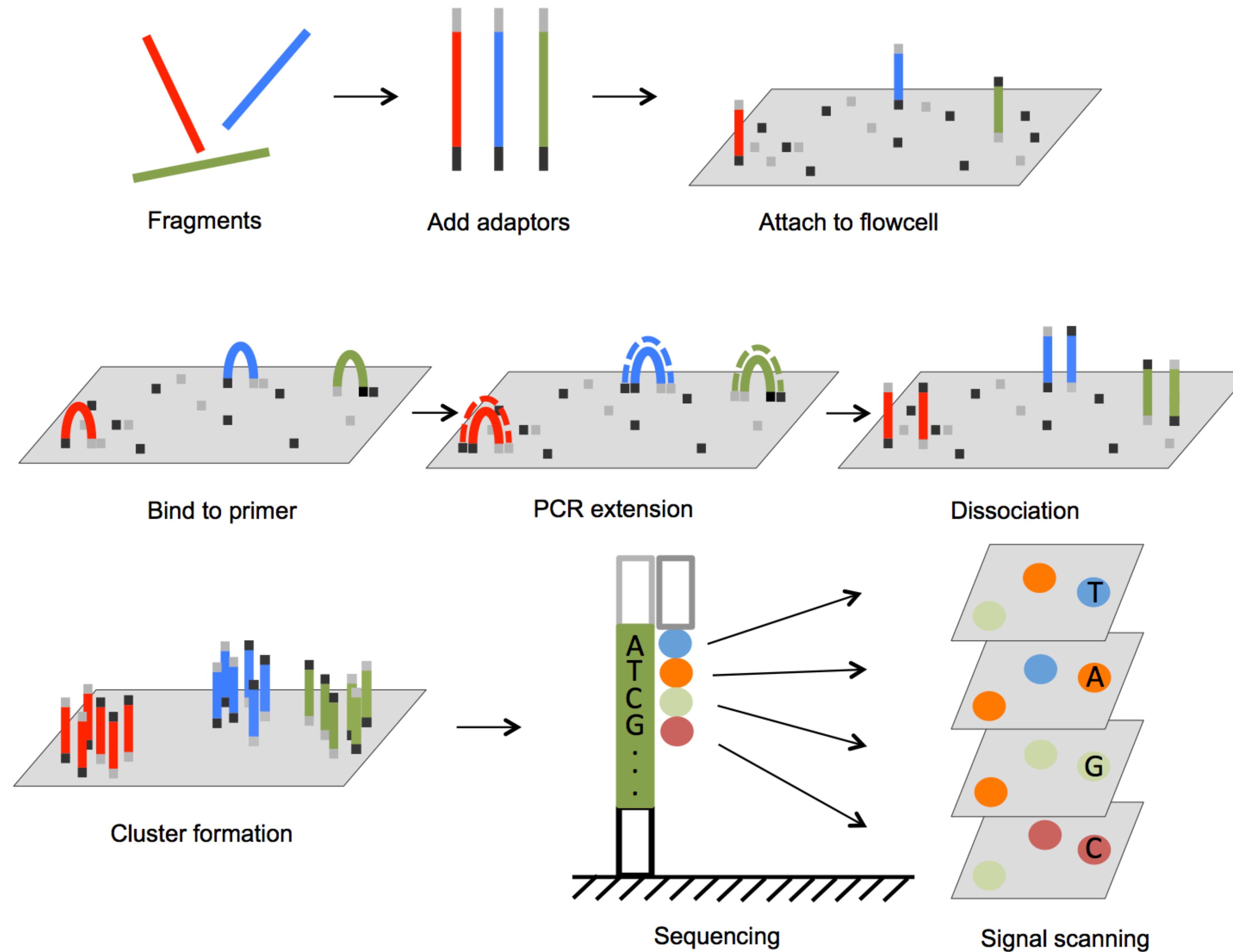
- ① Reaction mixture
    - ▶ Primer and DNA template
    - ▶ ddNTPs with flourochromes
    - ▶ DNA polymerase
    - ▶ dNTPs (dATP, dCTP, dGTP, and dTTP)



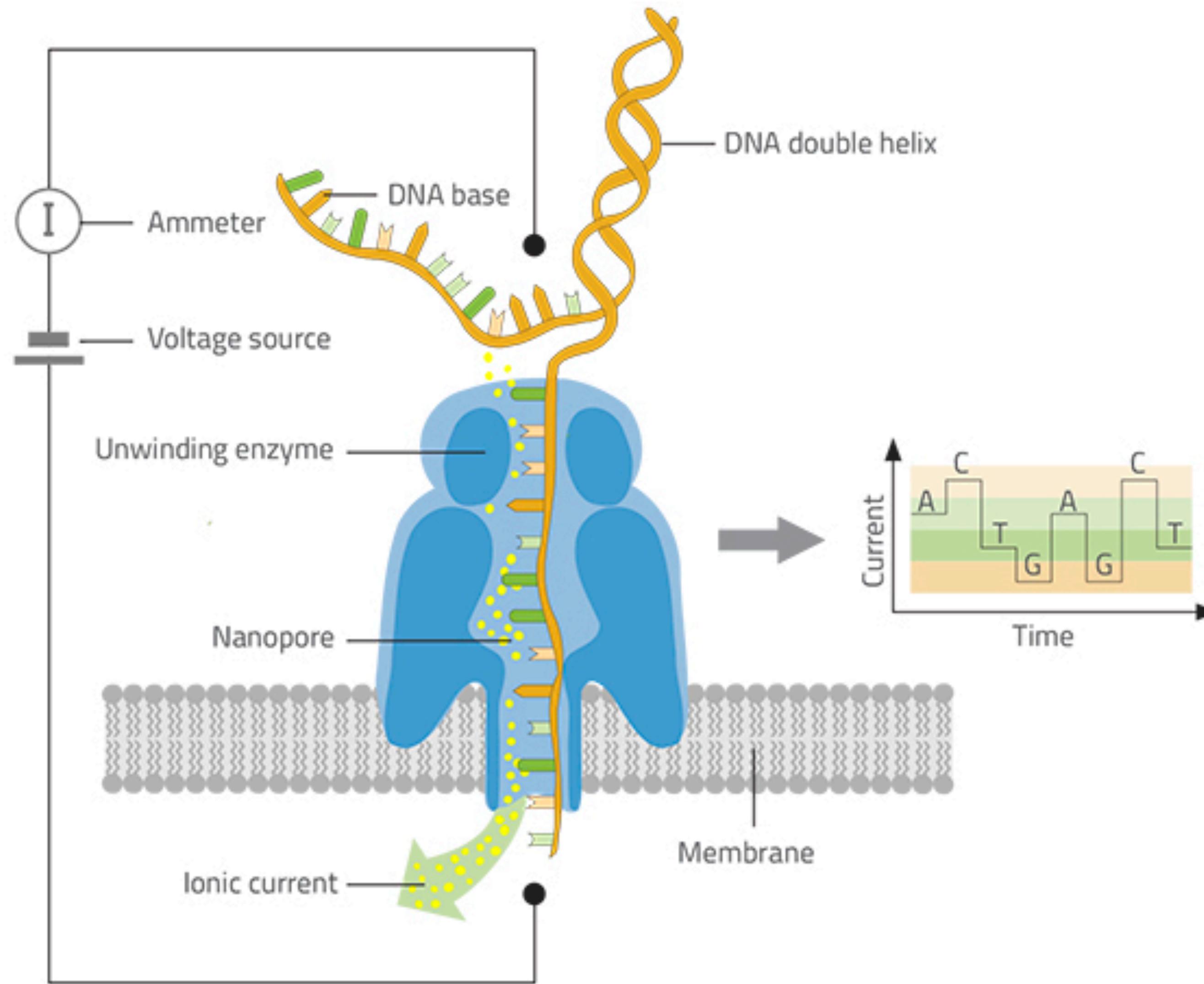
### ③ Capillary gel electrophoresis separation of DNA fragments



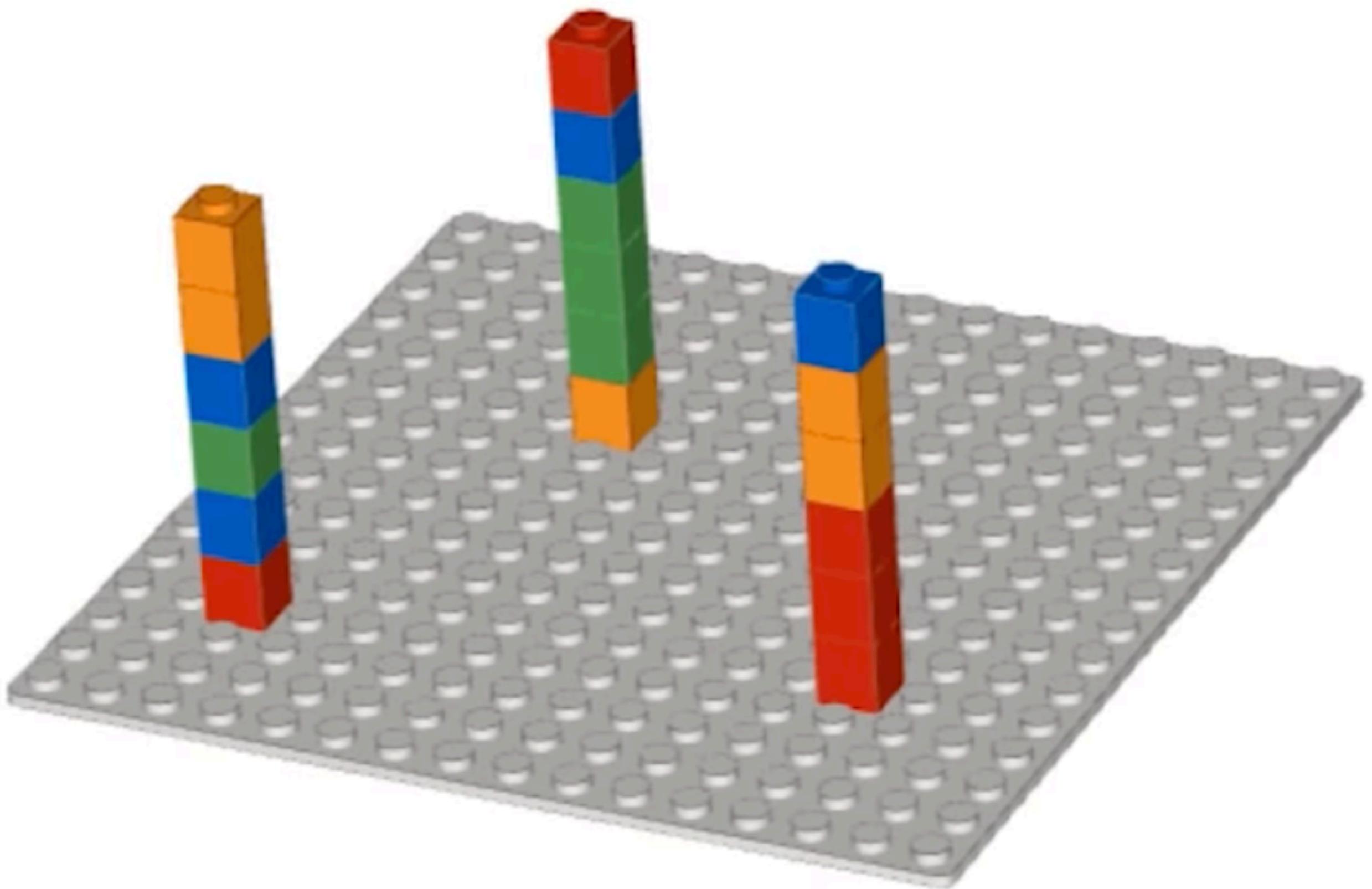
# Методы: Illumina



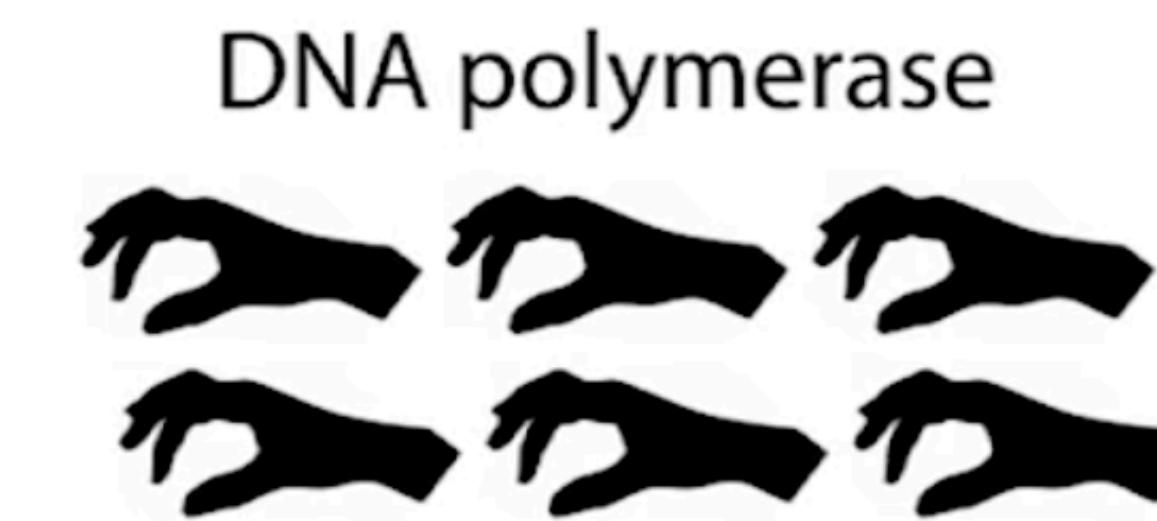
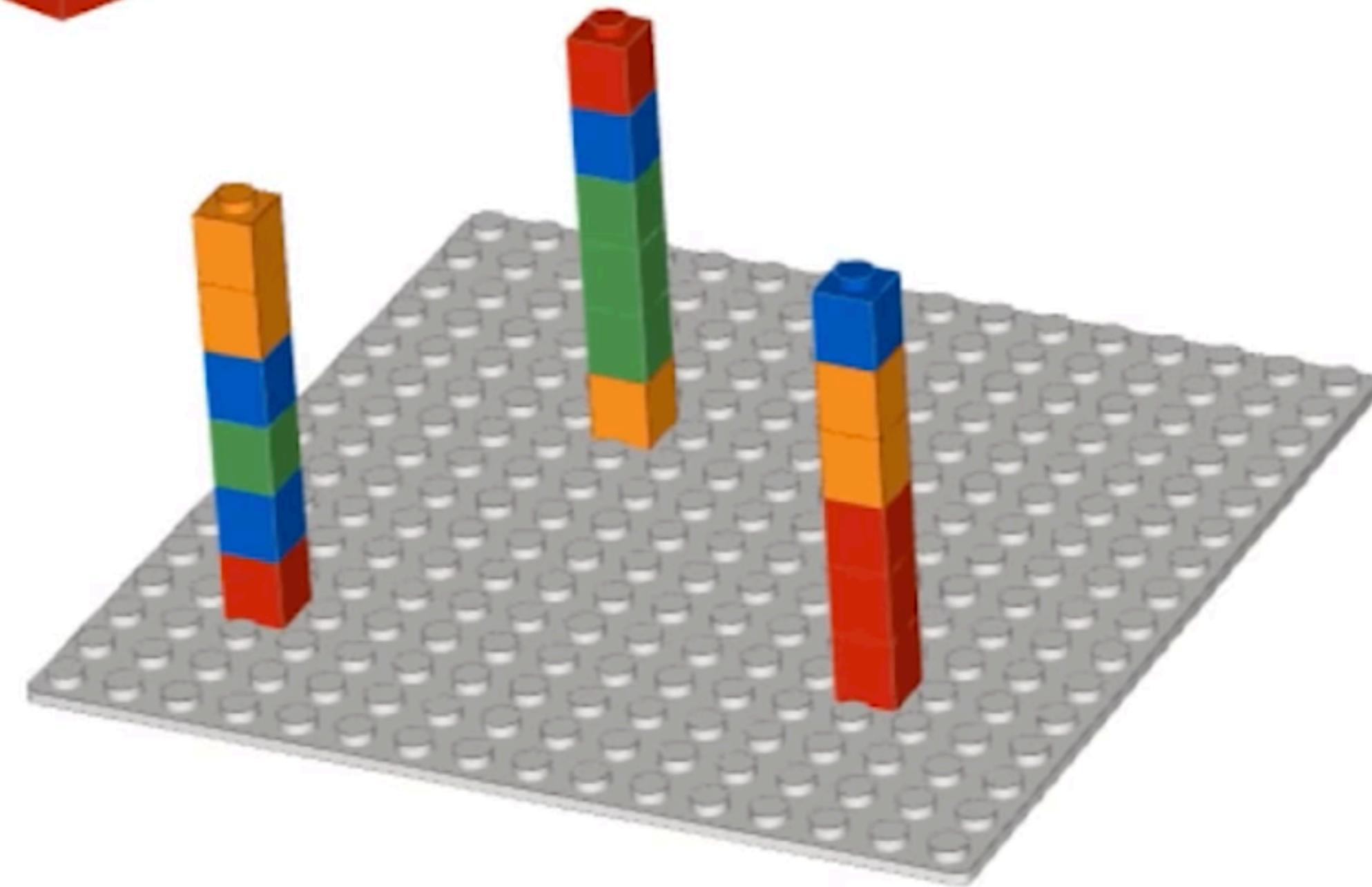
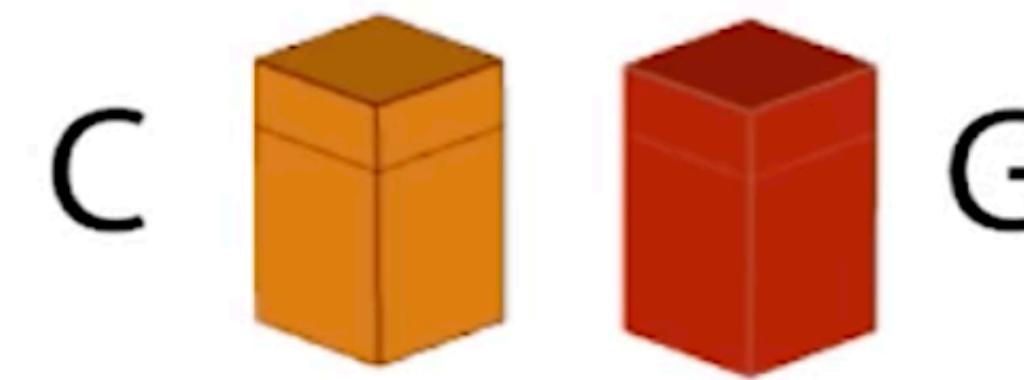
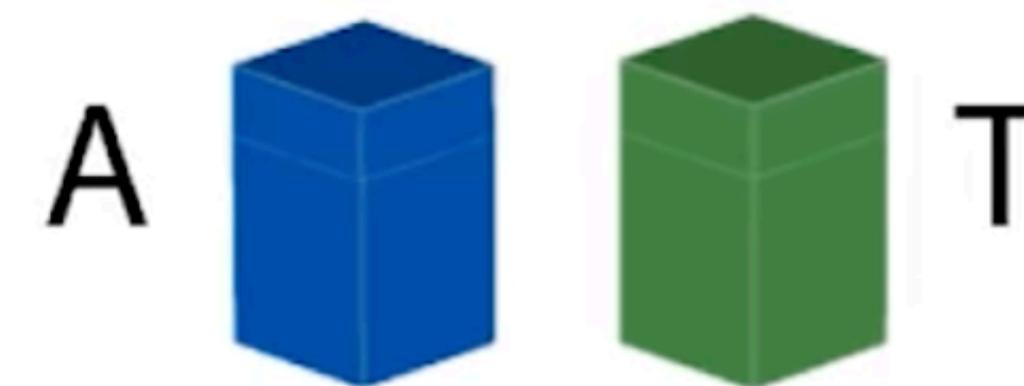
# Методы: Nanopore



# Illumina, модель

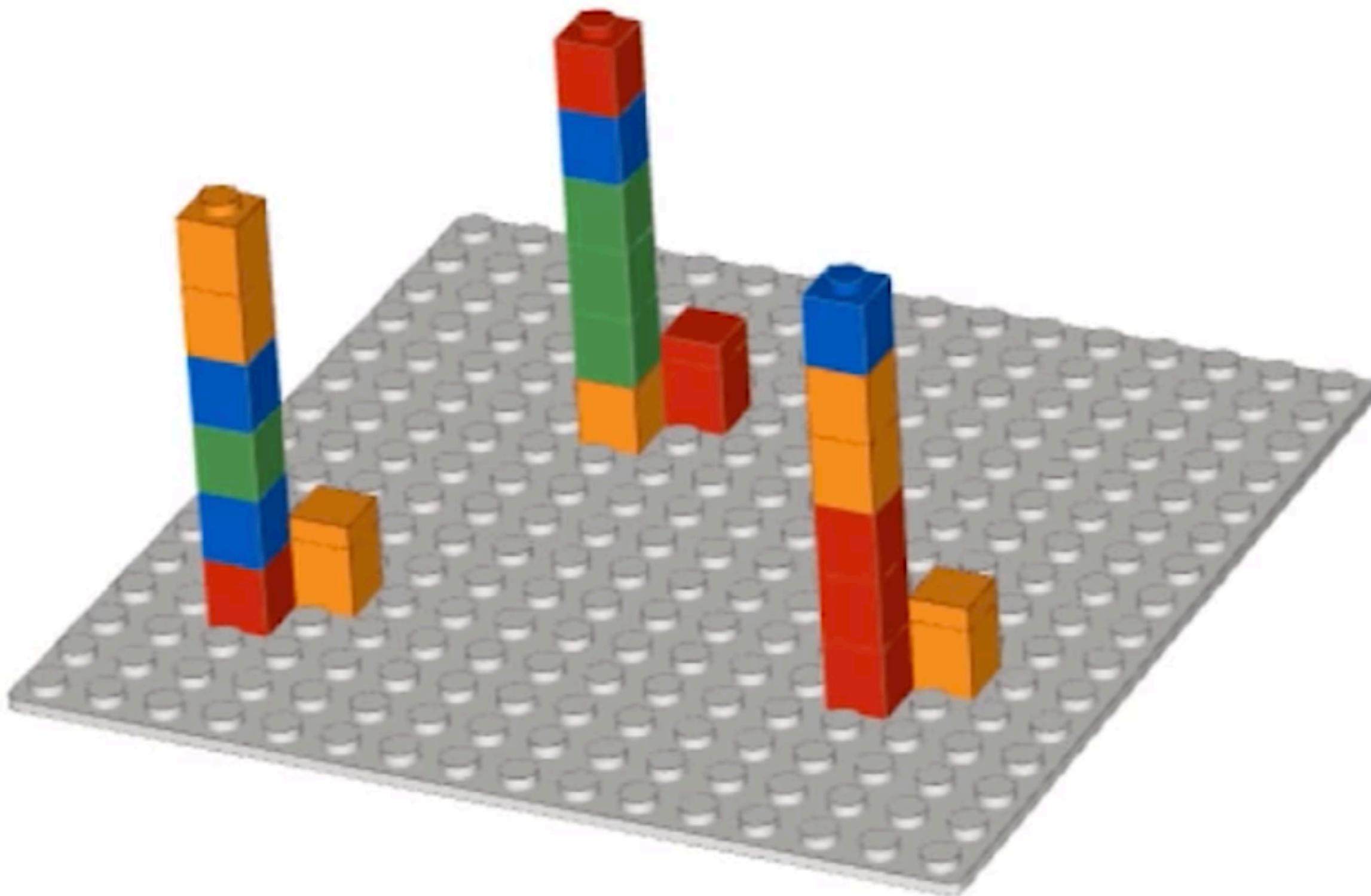


# Illumina, модель

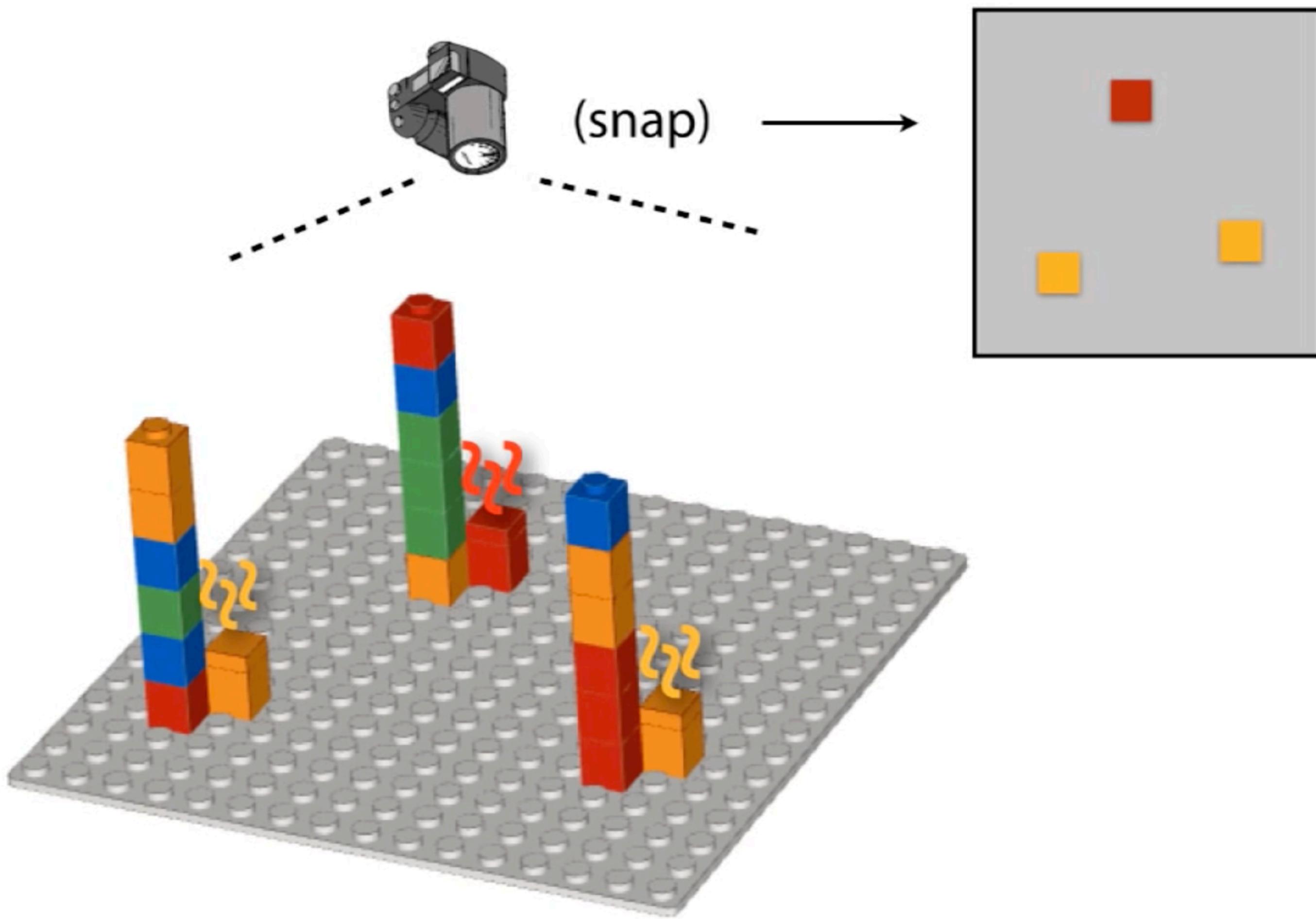


DNA polymerase

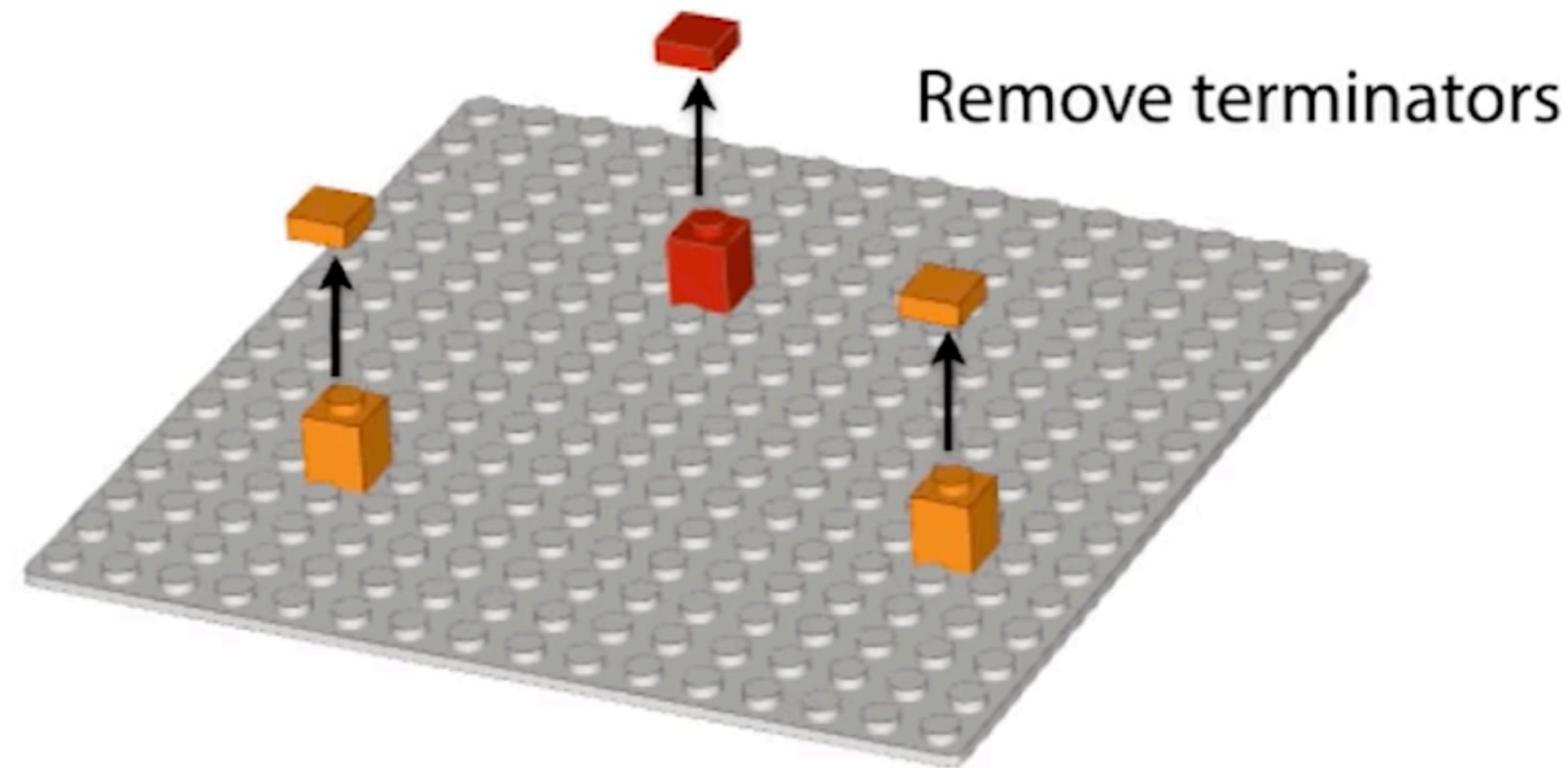
# Illumina, модель



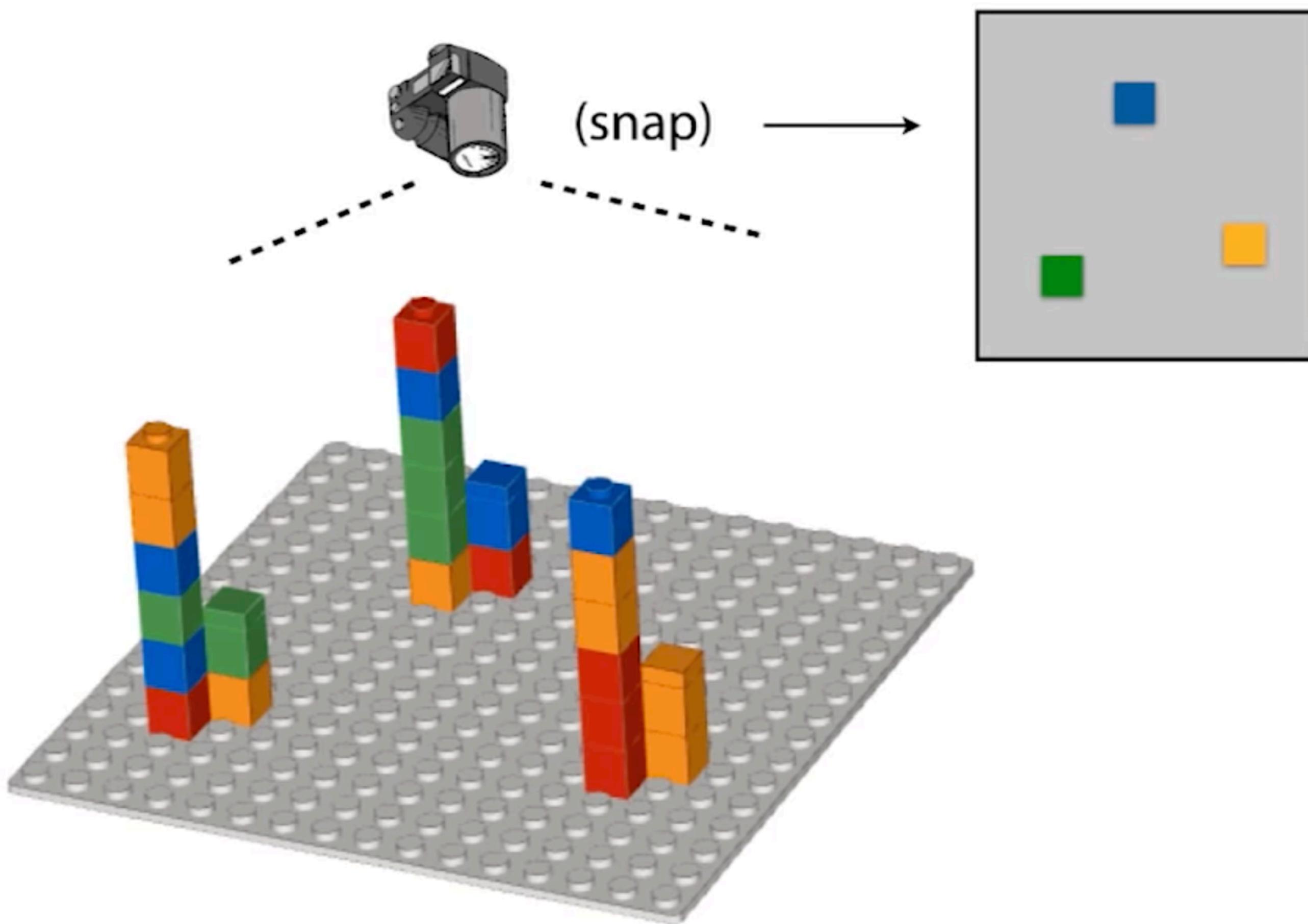
# Illumina, модель



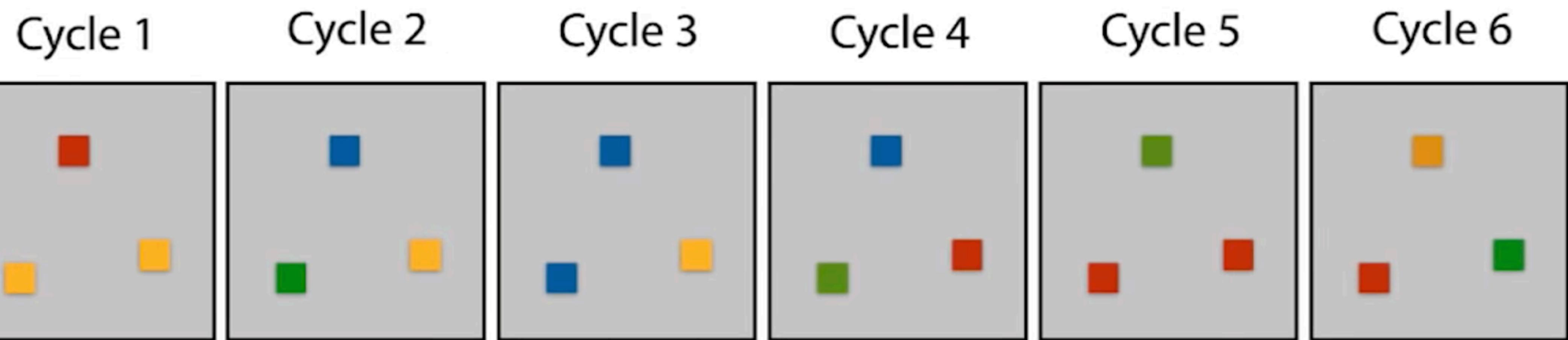
# Illumina, модель



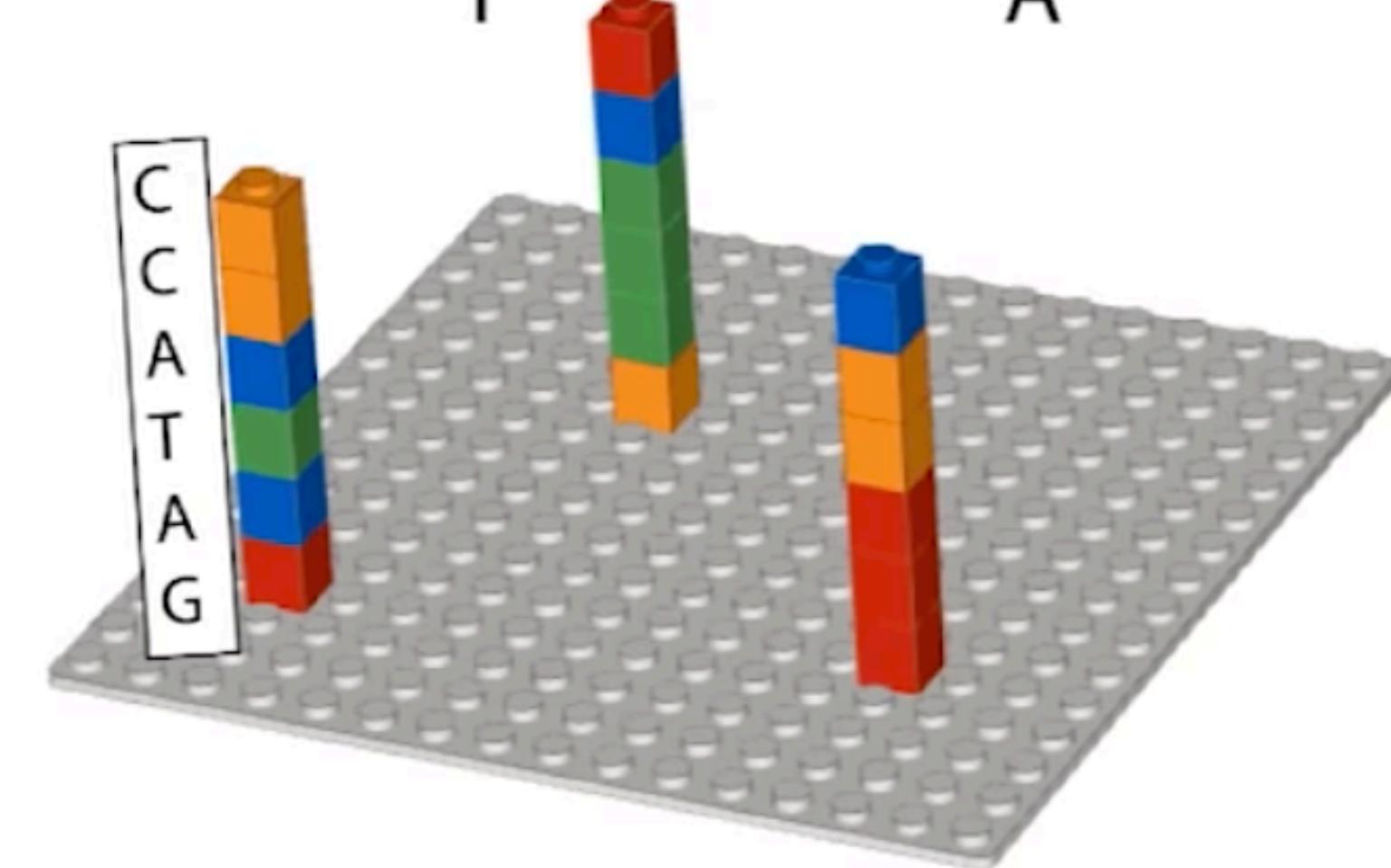
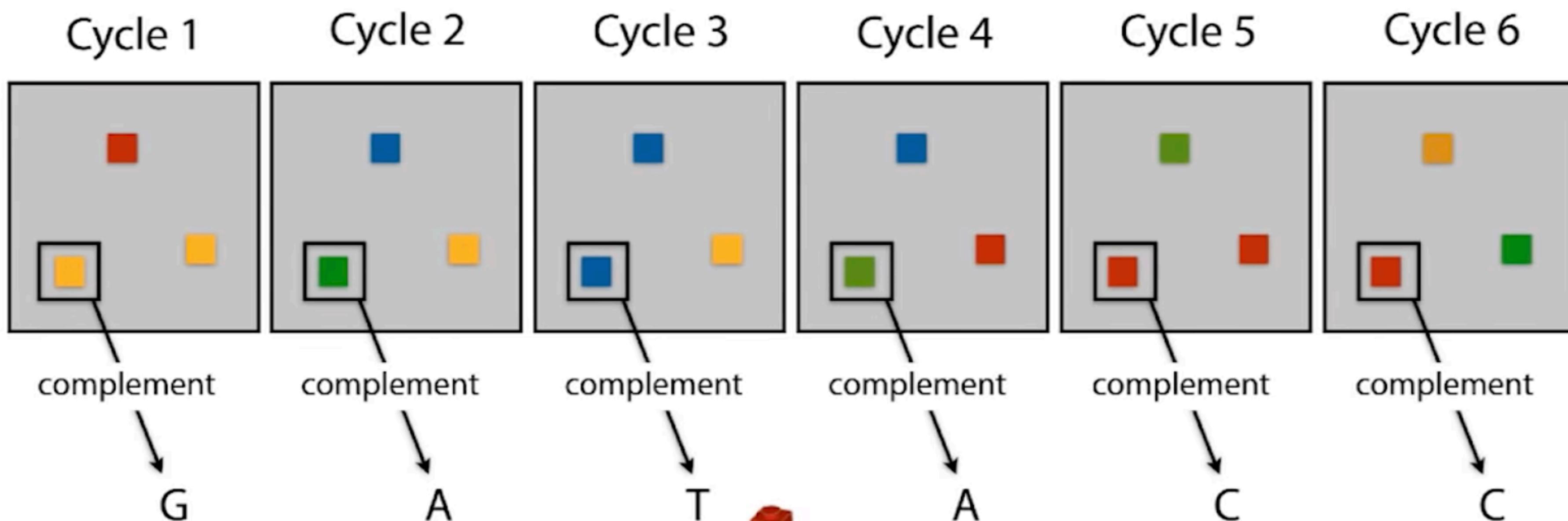
# Illumina, модель



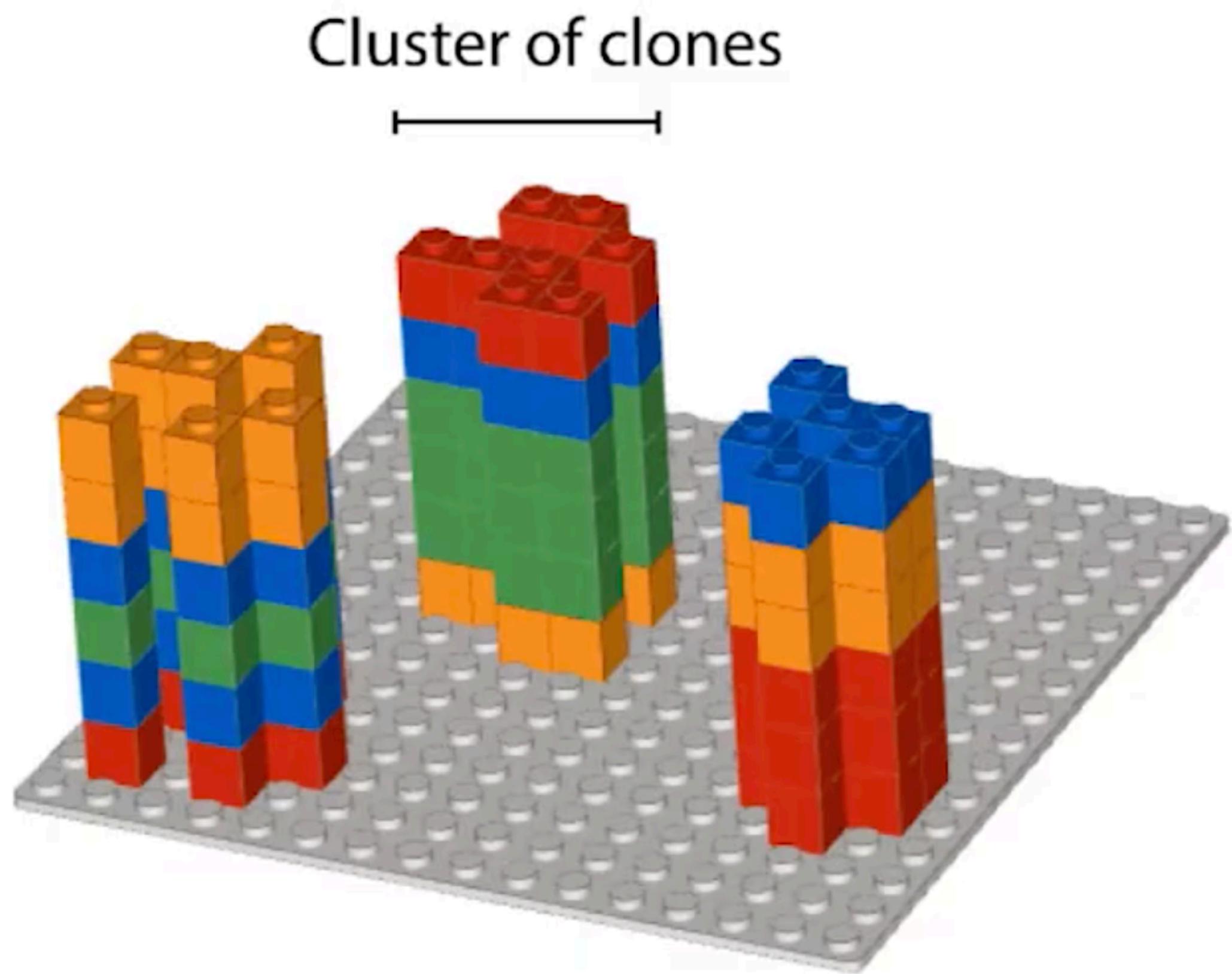
# Illumina, модель



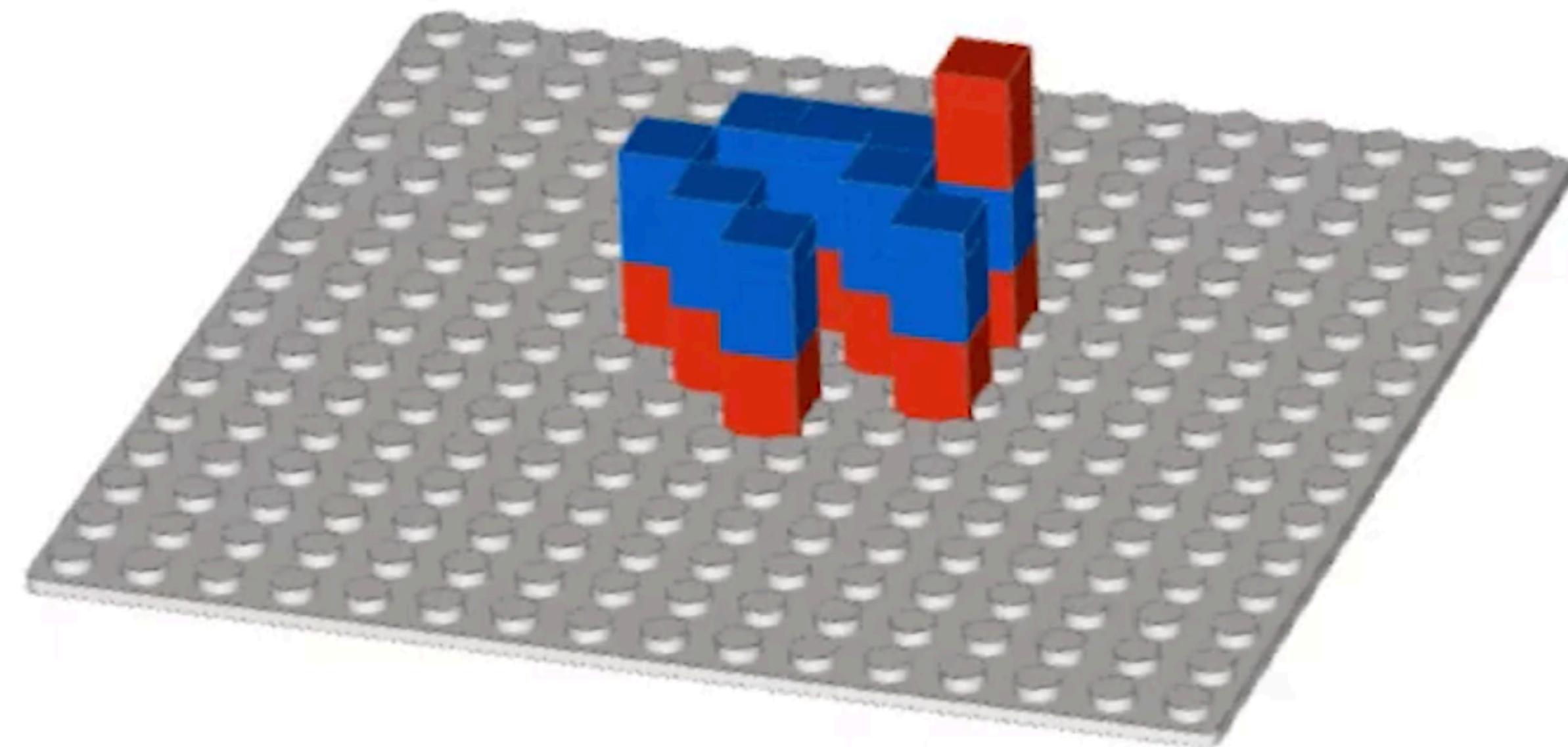
# Illumina, модель



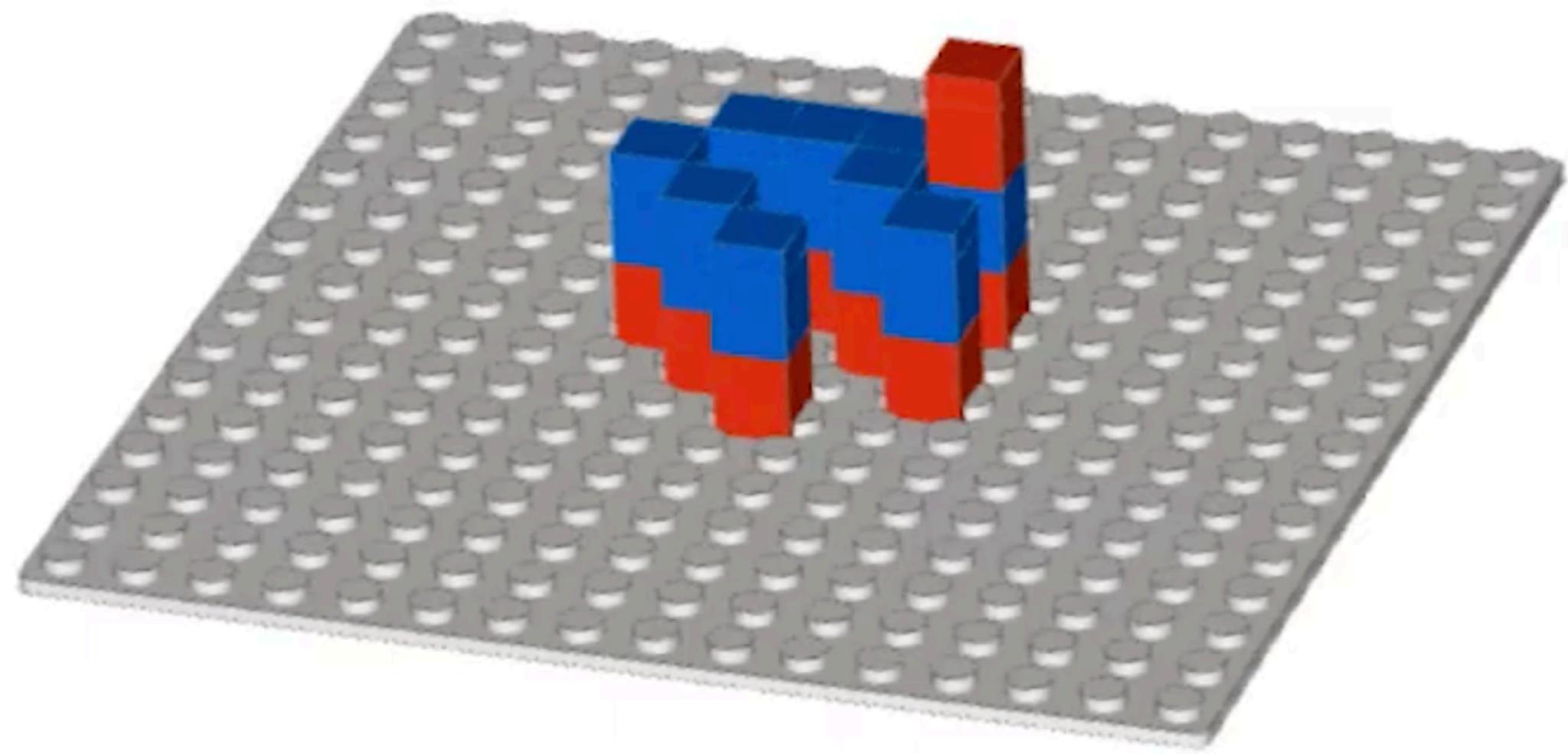
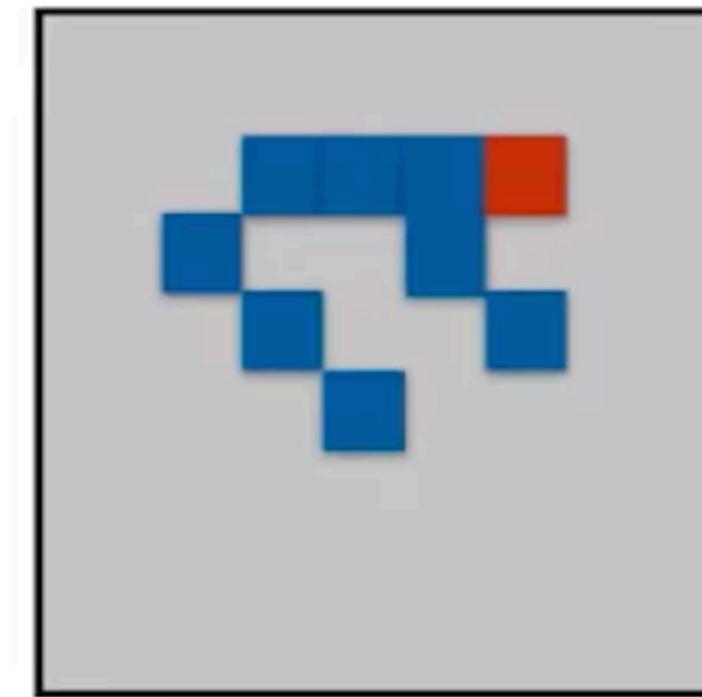
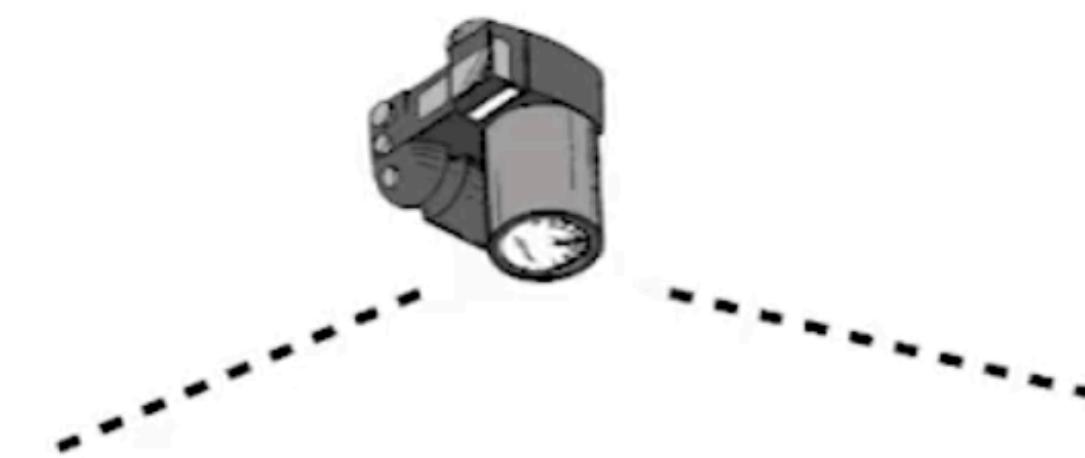
# Ошибки, на примере Illumina



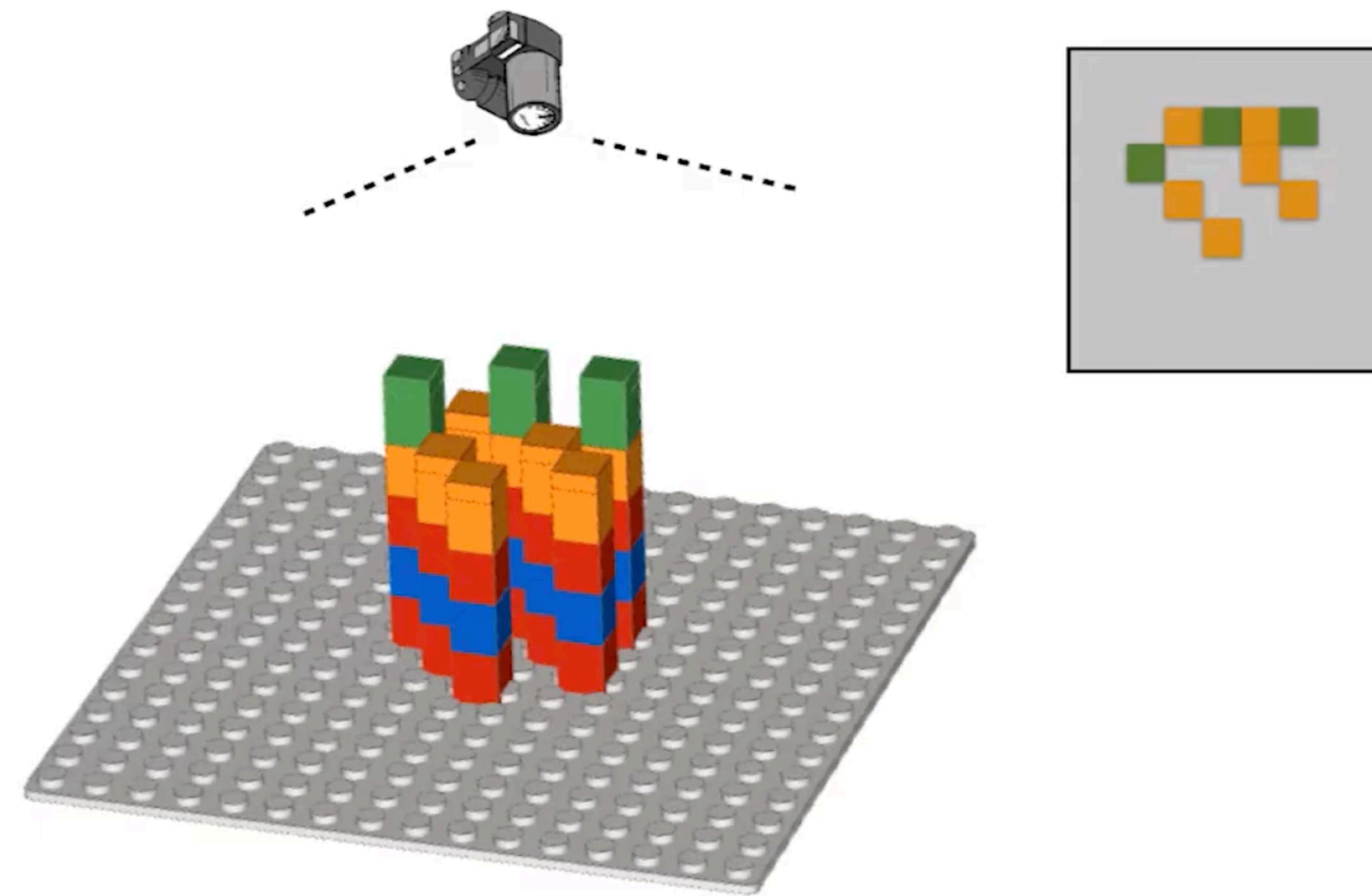
# Ошибки, на примере Illumina



# Ошибки, на примере Illumina



# Ошибки, на примере Illumina



# Ошибки, на примере Illumina

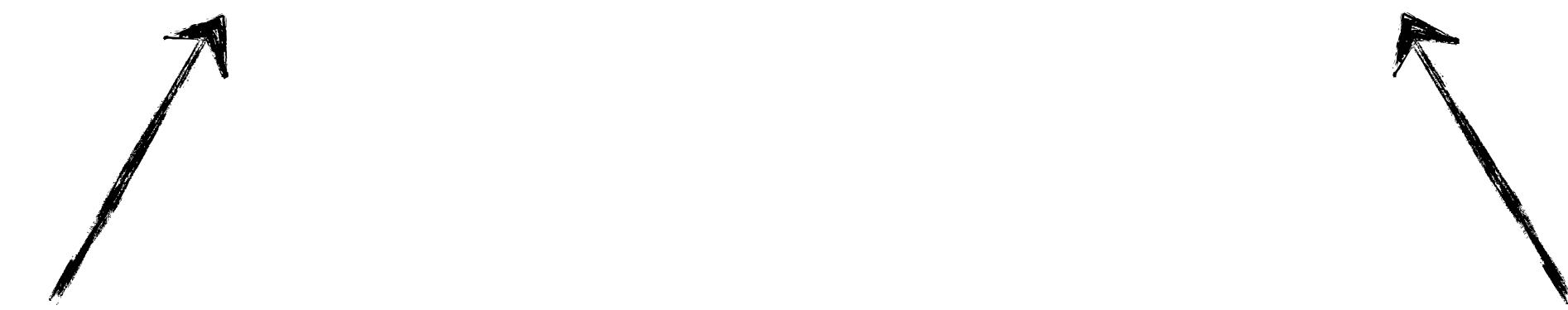
$$Q = -10 \log_{10}(p)$$

Качество прочтения

Вероятность ошибки

# Ошибки, на примере Illumina

$$Q = -10 \log_{10}(p)$$



Качество прочтения

Вероятность ошибки

$Q = 10 \rightarrow 1 \text{ к } 10$  что произошла ошибка

$Q = 20 \rightarrow 1 \text{ к } 100$

$Q = 30 \rightarrow 1 \text{ к } 1000$

# Ошибки, на примере Illumina



$$p = \frac{3}{9} = \frac{1}{3}$$

# Ошибки, на примере Illumina



$$p = \frac{3}{9} = \frac{1}{3}$$

$$Q = -10 \log_{10} \left( \frac{1}{3} \right) = 4.77$$

# FASTQ

```
@SEQ_ID
GATTGGGGTCAAAGCAGTATCGATCAAATAGTAAATCCATTGTTCAACTCACAGTT
+
! ' ' *((((***+))%%%++)(%%%%).1***-+*' ')**)55CCF>>>>CCCCCCCC65
```

# FASTQ

@SEQ\_ID [идентификатор]  
GATTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTGTTCAACTCACAGTT  
+  
! ' ' \*((((\*++))%%++)(%%%).1\*\*\*-+\*'')\*\*55CCF>>>>CCCCCCCC65

# FASTQ

@SEQ\_ID [идентификатор]

GATTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTGTTCAACTCACAGTT [последовательность]

+

! ' ' \*((((\*++))%%++)(%%%).1\*\*\*-+\*'')\*\*)55CCF>>>>CCCCCCC65

# FASTQ

@SEQ\_ID [идентификатор]

GATTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTGTTCAACTCACAGTT [последовательность]

+ [необязательная строка]

! ' \*(((\*\*\*+))%%++)(%%%).1\*\*\*-+\*' ))\*\*55CCF>>>>CCCCCCCC65

# FASTQ

@SEQ\_ID [идентификатор]  
GATTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTGTTCAACTCACAGTT [последовательность]  
+ [необязательная строка]  
! ' '\*((((\*\*\*+))%%++)(%%%).1\*\*\*-+\*'')\*\*55CCF>>>>CCCCCCCC65 [качество прочтения]

# FASTQ

@SEQ\_ID [идентификатор]  
GATTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTGTTCAACTCACAGTT [последовательность]  
+ [необязательная строка]  
! ' '\*((((\*\*\*+))%%++)(%%%).1\*\*\*-+\*'')\*\*55CCF>>>>CCCCCCCC65 [качество прочтения]

ASCII представление того самого  $Q$

Из качества в символ: `chr(Q + 33)`

Обратно: `ord(qual) - 33`

# Визуализация

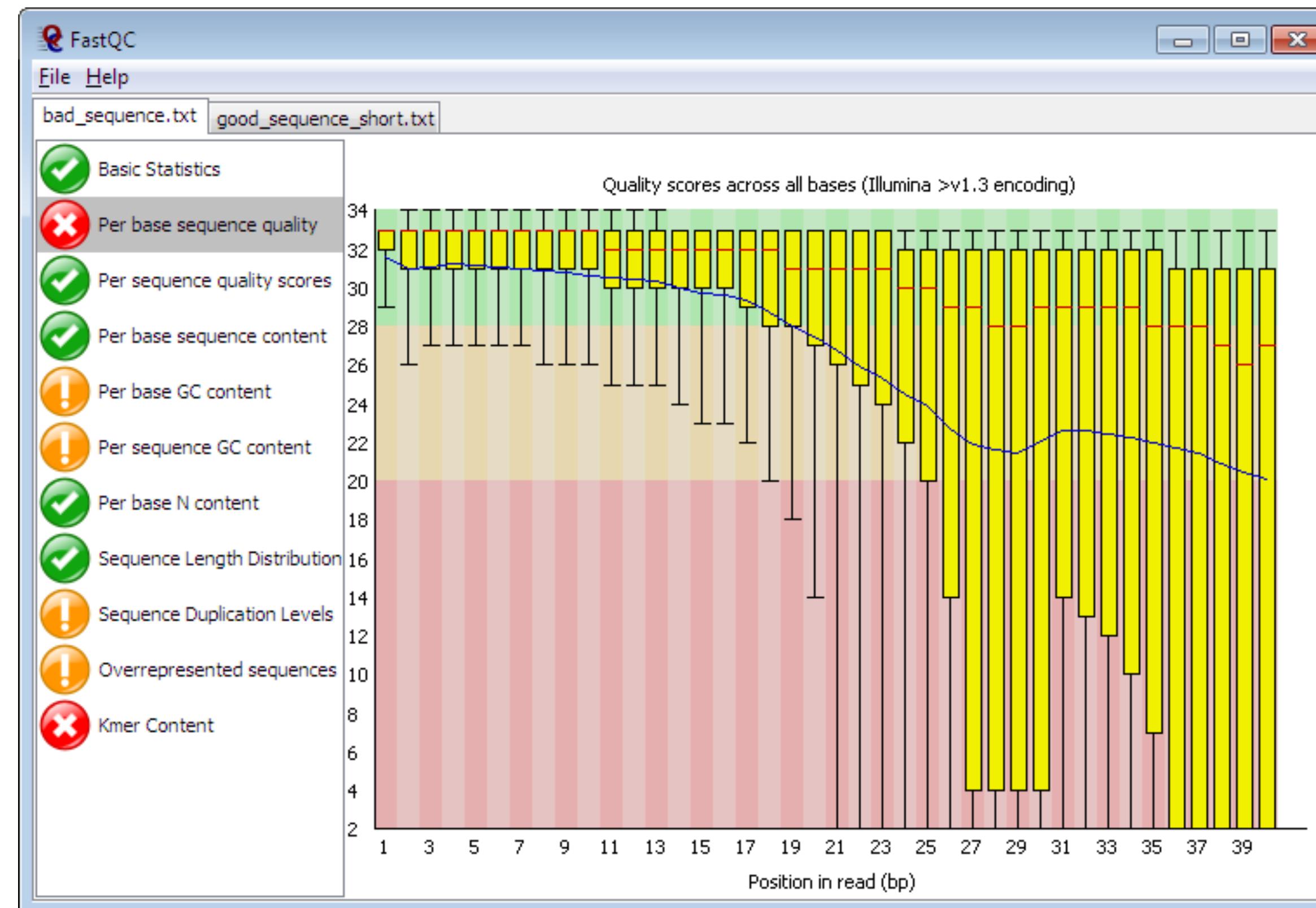
**FastQC** [<https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>]

```
>>fastqc bad_sequence.txt good_sequence.txt
```

# Визуализация

**FastQC** [<https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>]

```
>>fastqc bad_sequence.txt good_sequence.txt
```



# Удаление плохих ридов

**Trimmomatic** [<http://www.usadellab.org/cms/?page=trimmomatic>]

```
java -jar trimmomatic-0.35.jar SE -phred33 input.fq.gz output.fq.gz  
ILLUMINACLIP:TruSeq3-SE:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
```

# Удаление плохих ридов

**Trimmomatic** [<http://www.usadellab.org/cms/?page=trimmomatic>]

```
java -jar trimmomatic-0.35.jar SE -phred33 input.fq.gz output.fq.gz  
ILLUMINACLIP:TruSeq3-SE:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
```

Удаление адаптеров (ILLUMINACLIP:TruSeq3-SE:2:30:10)

# Удаление плохих ридов

**Trimmomatic** [<http://www.usadellab.org/cms/?page=trimmomatic>]

```
java -jar trimmomatic-0.35.jar SE -phred33 input.fq.gz output.fq.gz  
ILLUMINACLIP:TruSeq3-SE:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
```

Удаление адаптеров (ILLUMINACLIP:TruSeq3-SE:2:30:10)

Удаление низкокачественных вначале (с качеством хуже 3) (LEADING:3)

# Удаление плохих ридов

**Trimmomatic** [<http://www.usadellab.org/cms/?page=trimmomatic>]

```
java -jar trimmomatic-0.35.jar SE -phred33 input.fq.gz output.fq.gz  
ILLUMINACLIP:TruSeq3-SE:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
```

Удаление адаптеров (ILLUMINACLIP:TruSeq3-SE:2:30:10)

Удаление низкокачественных в начале (с качеством хуже 3) (LEADING:3)

Удаление низкокачественных в конце (с качеством хуже 3) (TRAILING:3)

# Удаление плохих ридов

**Trimmomatic** [<http://www.usadellab.org/cms/?page=trimmomatic>]

```
java -jar trimmomatic-0.35.jar SE -phred33 input.fq.gz output.fq.gz  
ILLUMINACLIP:TruSeq3-SE:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
```

Удаление адаптеров (ILLUMINACLIP:TruSeq3-SE:2:30:10)

Удаление низкокачественных в начале (с качеством хуже 3) (LEADING:3)

Удаление низкокачественных в конце (с качеством хуже 3) (TRAILING:3)

Сканировать окном в 4 нуклеотида, если среднее качество в окне ниже 15, то удалять  
(SLIDINGWINDOW:4:15)

Удалять риды короче 36 нуклеотидов (MINLEN:36)

# Исправление ошибок

Идея!

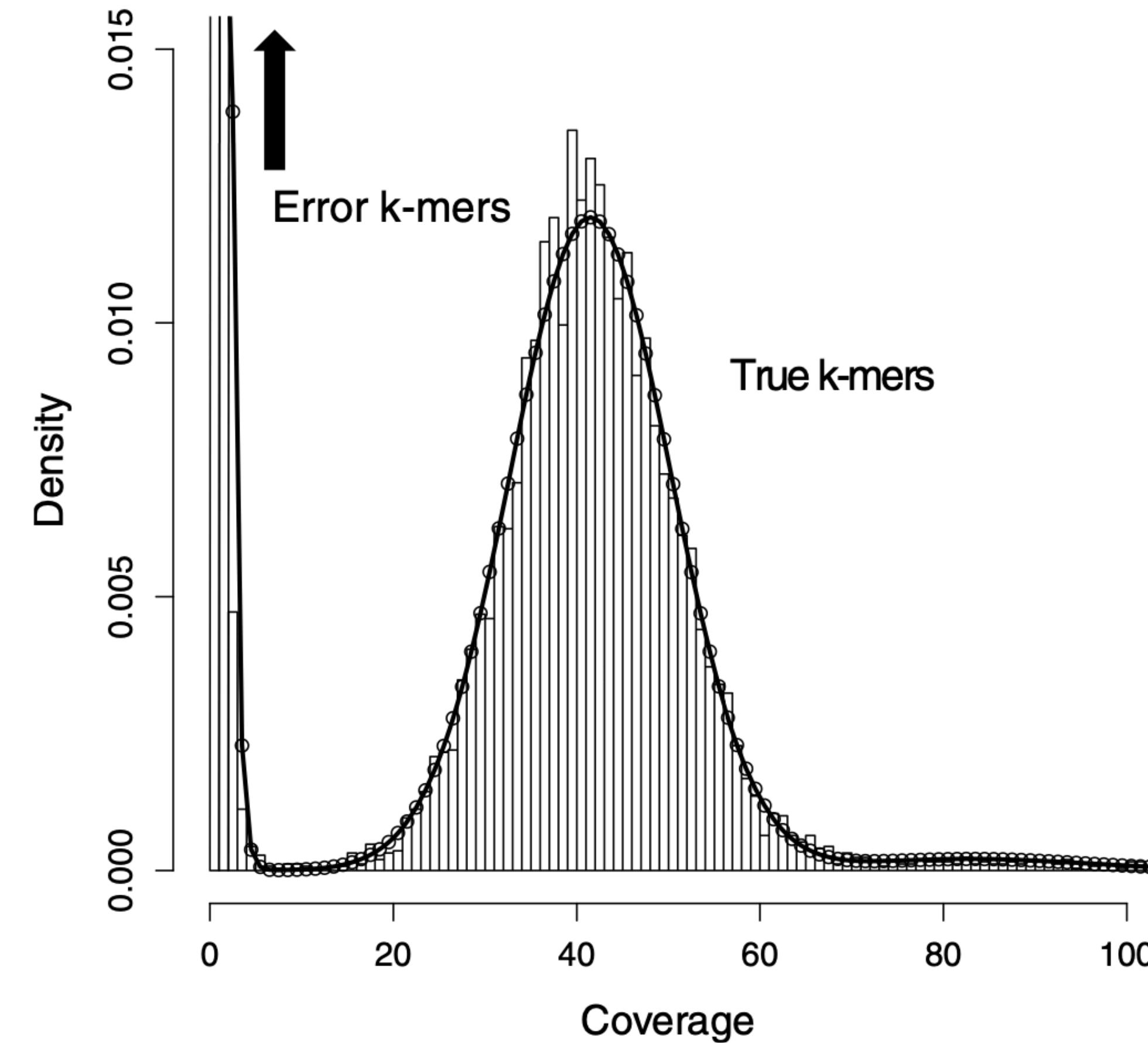
GATTTGGGTTCAAAGCAGTATCGATCAAATA  
GATTTGGGTTCAAAGCAGTATCGATCAAATA  
GATTTGGGTTCAAAGCAGTATCGATCAAATA  
GATTTG~~T~~GGGTTCAAAGCAGTATCGATCAAATA -> GATTTGG~~G~~GGGTTCAAAGCAGTATCGATCAAATA  
GATTTGGGTTCAAAGCAGTATCGATCAAATA  
GATTTGGGTTCAAAGCAGTATCGATCAAATA  
GATTTGGGTTCAAAGCAGTATCGATCAAATA  
GATTTGGGTTCAAAGCAGTATCGAT~~A~~AAATA -> GATTTGGGTTCAAAGCAGTATCGAT~~C~~AAATA  
GATTTGGGTTCAAAGCAGTATCGATCAAATA

# Исправление ошибок: Quake

1. Нужно посчитать все  $k$ -меры и определить, какие ошибочные
  - $k$ -мер: его количество не учитывает качество
  - Будем считать  $q$ -меры.  $k$ -мер: его количество \* произведение качеств

# Исправление ошибок: Quake

1. Нужно посчитать все  $k$ -меры и определить, какие ошибочные
  - $k$ -мер: его количество не учитывает качество
  - Будем считать  $q$ -меры.  $k$ -мер: его количество \* произведение качеств



# Исправление ошибок: Quake

1. Нужно посчитать все  $k$ -меры и определить, какие ошибочные
  - $k$ -мер: его количество не учитывает качество
  - Будем считать  $q$ -меры.  $k$ -мер: его количество \* произведение качеств
  - По распределению разделим  $k$ -меры на 2 кластера (ошибочные и правильные)
2. Все ошибочные  $k$ -меры – кандидаты на исправление в ридах
  - Нуклеотиды которые наблюдаем в ряде  $O = O_1, O_2, \dots, O_N$
  - Те которые на самом деле в геноме  $A = A_1, A_2, \dots, A_N$
  - Нам бы хотелось по наблюдаемым понять наиболее вероятные  $A_i$

# Исправление ошибок: Quake

1. Нужно посчитать все k-меры и определить, какие ошибочные
    - k-мер: его количество не учитывает качество
    - Будем считать q-меры. k-мер: его количество \* произведение качеств
    - По распределению разделим k-меры на 2 кластера (ошибочные и правильные)
  2. Все ошибочные k-меры – кандидаты на исправление в ридах
    - Нуклеотиды которые наблюдаем в ряде  $O = O_1, O_2, \dots, O_N$   
Те которые на самом деле в геноме  $A = A_1, A_2, \dots, A_N$   
Нам бы хотелось по наблюдаемым понять наиболее вероятные  $A_i$
    -
- $$P(A = a | O = o) = \prod_{i=1}^N \frac{P(O_i = o_i | A_i = a_i)P(A_i = a_i)}{P(O_i = o_i)}$$

# Исправление ошибок: Quake

1. Нужно посчитать все k-меры и определить, какие ошибочные
  - k-мер: его количество не учитывает качество
  - Будем считать q-меры. k-мер: его количество \* произведение качеств
  - По распределению разделим k-меры на 2 кластера (ошибочные и правильные)
2. Все ошибочные k-меры – кандидаты на исправление в ридах
  - Нуклеотиды которые наблюдаем в ряде  $O = O_1, O_2, \dots, O_N$   
Те которые на самом деле в геноме  $A = A_1, A_2, \dots, A_N$   
Нам бы хотелось по наблюдаемым понять наиболее вероятные  $A_i$
  - $$P(A = a | O = o) = \prod_{i=1}^N \frac{P(O_i = o_i | A_i = a_i)P(A_i = a_i)}{P(O_i = o_i)}$$
  - $$P(O_i = o_i | A_i = a_i) = \begin{cases} p_i & \text{if } o_i = a_i \\ (1-p_i)E_{q_i}(a_i, o_i) & \text{otherwise} \end{cases}$$
 где  $p_i = 1 - 10^{-\frac{q_i}{10}}$

# Исправление ошибок: Quake

1. Нужно посчитать все k-меры и определить, какие ошибочные
  - k-мер: его количество не учитывает качество
  - Будем считать q-меры. k-мер: его количество \* произведение качеств
  - По распределению разделим k-меры на 2 кластера (ошибочные и правильные)
2. Все ошибочные k-меры – кандидаты на исправление в ридах
  - Нуклеотиды которые наблюдаем в ряде  $O = O_1, O_2, \dots, O_N$   
Те которые на самом деле в геноме  $A = A_1, A_2, \dots, A_N$   
Нам бы хотелось по наблюдаемым понять наиболее вероятные  $A_i$
  - $$P(A = a | O = o) = \prod_{i=1}^N \frac{P(O_i = o_i | A_i = a_i)P(A_i = a_i)}{P(O_i = o_i)}$$
  - $$P(O_i = o_i | A_i = a_i) = \begin{cases} p_i & \text{if } o_i = a_i \\ (1-p_i)E_{q_i}(a_i, o_i) & \text{otherwise} \end{cases}$$
 где  $p_i = 1 - 10^{-\frac{q_i}{10}}$
3. Коррекция

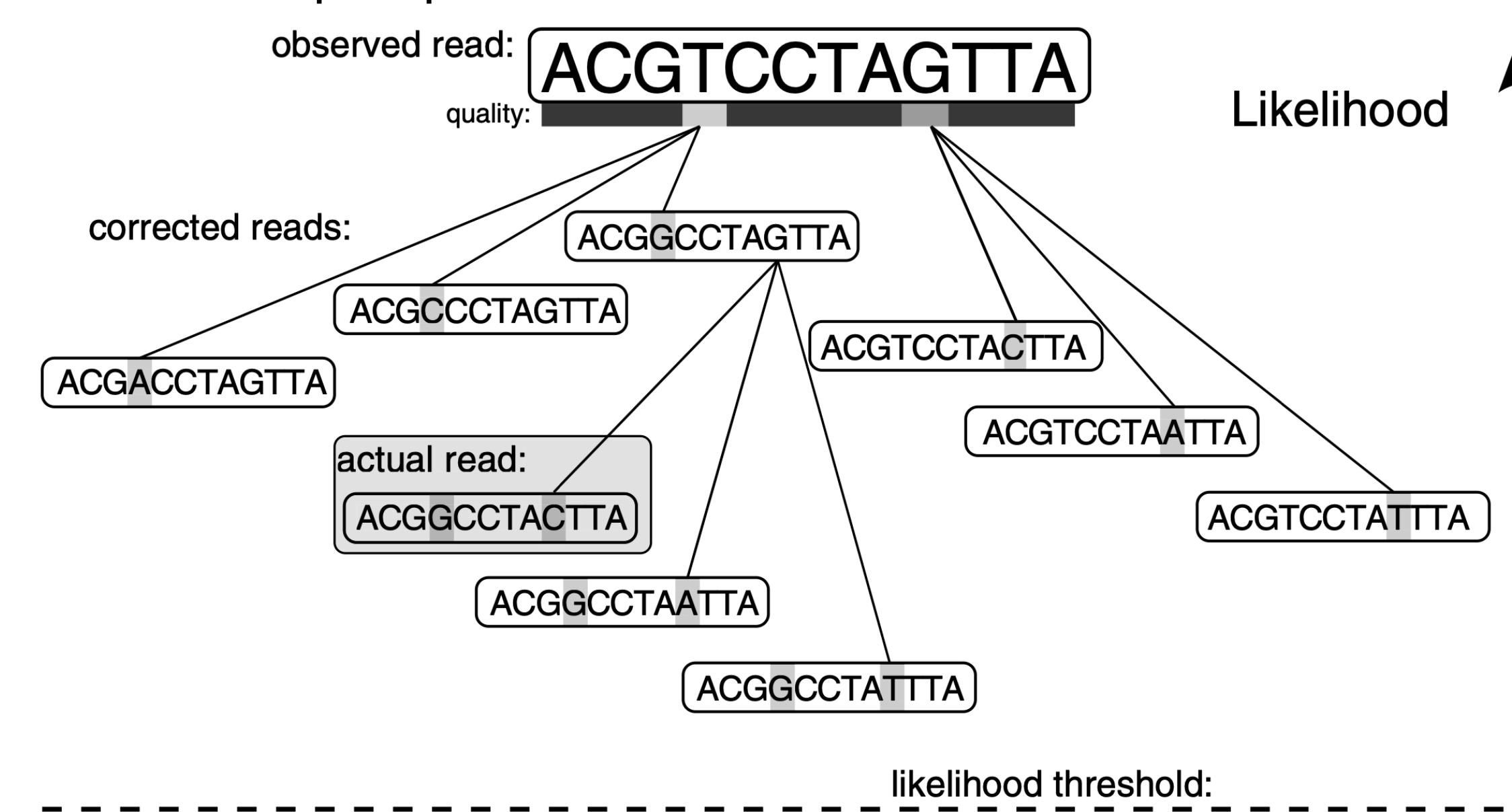
# Исправление ошибок: Quake

1. Нужно посчитать все  $k$ -меры и определить, какие ошибочные
2. Все ошибочные  $k$ -меры – кандидаты на исправление в ридах
3. Коррекция
  - о Хотим сделать в каждом ряде такие замены, которые переместят все  $k$ -меры из кластера ошибочных в кластер правильных
  - о 

```
1: function SEARCH(R)
2:   P.PUSH({} , 1)
3:   while (C, L) ← P.pop() do
4:     if VALID (R, C) then
5:       return C
6:     else
7:       i ← lowest quality unconsidered position
8:       for nt ∈ [A, C, G, T] do
9:         if R[i] == nt then
10:           Cnt = C
11:         else
12:           Cnt = C + ( i, nt )
13:           Lnt ← LIKELIHOODRATIO (R, Cnt)
14:           if Lnt > likelihood_threshold then
15:             P.push(Cnt, Lnt)
16:   return {}
```

# Исправление ошибок: Quake

1. Нужно посчитать все  $k$ -меры и определить, какие ошибочные
2. Все ошибочные  $k$ -меры – кандидаты на исправление в ридах
3. Коррекция
  - Хотим сделать в каждом риде такие замены, которые переместят все  $k$ -меры из кластера ошибочных в кластер правильных
  -



# Исправление ошибок: Quake

**Quake:** [<http://www.cbcn.umd.edu/software/quake/manual.html>]  
->quake.py -f [fastq file list] -k [k-mer size] -p 4

# Резюмируем

- Секвенирование – случайный процесс
- Могут происходить ошибки секвенирования
- Можно просто отбрасывать риды с ошибками
- Но можно и исправлять!