

# Выравнивание биологических последовательностей. Программа BLAST.

С.А. Спирин

[sas@belozersky.msu.ru](mailto:sas@belozersky.msu.ru)

МФК "Биоинформатика", 6 мая 2020



## Последовательности миоглобинов человека, мыши и быка

>MYG\_HUMAN

MGLSDGEWQLVLNVWGKVEADIPGHGQEVLI RLFK GHPETLEKFDKFKHLKSEDEMKASE  
DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH  
PGDFGADAQGAMNKALELFRKDMASNYKELGFQG

>MYG\_MOUSE

MGLSDGEWQLVLNVWGKVEADLAGHGQEVLI GLFKTHPETLDKFDKFKNLKSEEDMKGSE  
DLKKHGCTVLTALGTILKKKGQHAAEIQLAQSHATKHKIPVKYLEFISEIIIIEVLKKRH  
SGDFGADAQGAMSKALELFRNDIAAKYKELGFQG

>MYG\_BOVIN

MGLSDGEWQLVLNAWGKVEADVAGHGQEVLI RLF TGH PETLEKFDKFKHLKTEAEMKASE  
DLKKHGNTVLTALGGILKKKGHHEAEVKHLAESHANKHKIPVKYLEFISDAIIHVLHAKH  
PSDFGADAQAAMSKALELFRNDMAAQYKVLGFHG



## Напишем последовательности друг под другом, чтобы было лучше видно сходство

```

MYG_HUMAN MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIIRLFKGHPETLEKFDKFKHLKSEDEMKASE 60
MYG_MOUSE MGLSDGEWQLVLNVWGKVEADLAGHGQEVLIIGLFKTHPETLDKFDKFKNLKSEEDMKGSE 60
MYG_BOVIN MGLSDGEWQLVLNNAWGKVEADVAGHGQEVLIIRLFTGHPETLEKFDKFKHLKTEAEMKASE 60
              *****.*****:***** ** . *****:*****:***:* :**.*
              ****

MYG_HUMAN DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH 120
MYG_MOUSE DLKKHGCTVLTALGTILKKKGQHA AEIQPLAQSHATKHKIPVKYLEFISEIIIEVLKKRH 120
MYG_BOVIN DLKKHGNTVLTALGGILKKKGHHEAEV KHLAESHANKHKIPVKYLEFISDAIIHVLHAKH 120
              ***** ***** *****:* **:** **:*** .*****:*****: **.*:**: :*

MYG_HUMAN PGDFGADAQGAMNKALELFRKDMASNYKELGFQG 154
MYG_MOUSE SGDFGADAQGAMSKALELFRNDIAAKYKELGFQG 154
MYG_BOVIN PPDFGADAQAAMSKALELFRNDMAAQYKVLGFHG 154
              .*****.** .*****:*:*:*:* **:**
  
```

Видно, что большинство букв совпадает, но некоторые различаются.

Это последовательности **гомологических** белков, что означает, что эти белки произошли от общего предка. За время, прошедшее от существования общего предка, некоторые буквы менялись, но большинство остались неизменными.



# Последовательности миоглобинов человека и рыбы

>MYG\_HUMAN

MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIIRLFKGHPEKLEKFDKFKHLKSEDEMKASE  
DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH  
PGDFGADAQGAMNKALELFRKDMASNYKELGFQG

>MYG\_DANRE

MADHDLVLKCGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQGDLAGSPAVAAN  
GATVLKKLGELLKAKGDHAALLKPLANTHANIHKVALNNFRLITEVLVKVMAEKAGLDAA  
QGALRRVMDAVIDIDGYYKEIGFAG

Разная длина, как сравнивать?

# Последовательности миоглобинов человека и рыбы

>MYG\_HUMAN

MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIIRLFKGHPEKFDKFKHLKSEDEMKASE  
DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH  
PGDFGADAQGAMNKALELFRKDMASNYKELGFQG

>MYG\_DANRE

MADHDLVLKCGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQGDLAGSPA  
VAH  
GATVLKKGELLLKAKGDHAALLKPLANTHANIHKVALNNFRLITEVLVKVMAEKAGL  
DAA  
QGALRRVMDAVIDIDGYYKEIGFAG

Разная длина, как сравнивать?

Ответ: **выравнивание**

```
MYG_HUMAN MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIIRLFKGHPEKFDKFKHLKSEDEMKASE 60
MYG_DANRE ----MADHDLVLKCGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQGDLAGSPA 55
          .: :***: ** ***** .:* *** ***** :*:** : * **. :.. :. *
```

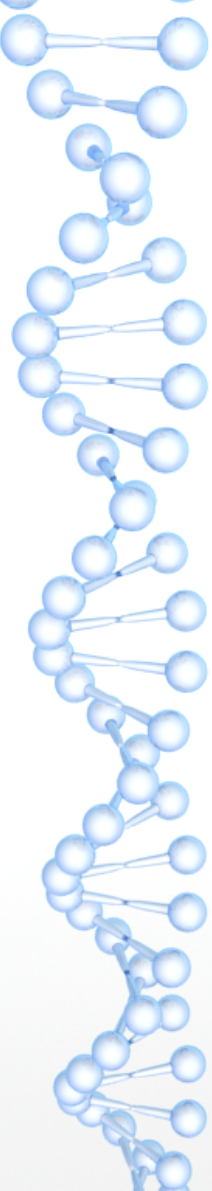
```
MYG_HUMAN DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH 120
MYG_DANRE AVAAHGATVLKKGELLLKAKGDHAALLKPLANTHANIHKVALNNFRLITEVLVKVMAEKA 115
          : ***** . ** :** ** . * :*****:*. **: :. :.***: :***: . *
```

```
MYG_HUMAN PGDFGADAQGAMNKALELFRKDMASNYKELGFQG 154
MYG_DANRE --GLDAAGQGALRRVMDAVIDIDGYYKEIGFAG 147
          .:. * .***:..: . * : . ***:*** *
```

# Последовательности миоглобинов человека и рыбы

**Гэпы** показывают, что для данного аминокислотного остатка **нет** гомологичного в другом белке

Биологическая причина — инсерции и делеции, закрепившиеся в эволюции.  
Отличить инсерцию от делеции мы не можем.



```
MYG_HUMAN MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIIRLFKGHPEKFDKFKHLKSEDEMKASE 60
MYG_DANRE ----MADHDLVLKCGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQG-DLAGSP 55
      .: :***: ** ***** .:* *** ***** :*:** : * **. ... :: .*

MYG_HUMAN DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH 120
MYG_DANRE AVAAHGATVLKKGELLLKAKGDHAALLKPLANTHANIHKVALNNFRLITEVLVKVMAEKA 115
      : ***** . ** :** ** . * :*****:***. **: :: :*:** :***: .*

MYG_HUMAN PGDFGADAQGAMNKALELFRKDMASNYKELGFQG 154
MYG_DANRE --GLDAAGQGALRRVMDAVIDGIDGYKKEIGFAG 147
      .:. * .***:..... . * : . ***:** *
```

# Выравнивание белков, гомологичных не по всей длине

```

ANTP_DROME  MTMSTNNCESMTSYFTNSYMGADMHHGHYPGNGVTDLDAQQMHHYSQN----ANHQGNMP 56
HXA1_HUMAN  -----MDNARMNSFLEYPILSSGDSGTCS 24
                                     :*  *:  :  :  :...*.

ANTP_DROME  YPRFPPYDRMPYYN-----GQGMDQQQHQVYSRPDSSQVGGVMPQ 99
HXA1_HUMAN  ARAYPSDHRITTFQSCAVSANS CGGDDRFLVGRGVQIGSPHHHHH----HHHPQPAT 79
          :*  .*:  ::
                                     *: *:  . *:  :  :  :

ANTP_DROME  AQTNGQLGVPQQQQQQQQPSQNQQQQQAQAPQQLQQQLPQVTQQVTHPQQQQQPVVY 159
HXA1_HUMAN  YQTSGNLGVSYSYSHSS--CGPSYGSQNF-----SAPY 108
          **.*:***  :...  **  .*:
                                     . *

ANTP_DROME  ASCKLQAAVGG LGMVPEGGSPPLVDQMSGHHMNAQMTLPHHMGHPQAQLGYTD--VGVPD 217
HXA1_HUMAN  SPYALNQEAD-----VSGGYPCAPAVYSGNLSSPMVQHH----HHHQGYAGGAVGSPQ 158
          :  * :  ..  .** *  .  :  :  :  :  *  *  :  :  ** :  *  *  * :

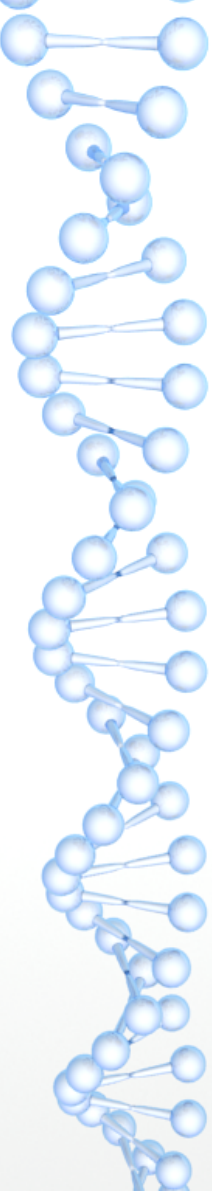
ANTP_DROME  VTE--VHQNHNNMGMYQQQSGVPPVGAPPQGMMHQGGPPQMHHQGHGHTPPSQNPNSQ 275
HXA1_HUMAN  YIHHSYGQEHQSLALATYNNSLSP-----LHASHQE----ACRSP-AS 196
          .  * : * : : : :  : : :  *  *  :  :  *  :

ANTP_DROME  SSGMPSPLYPW MRSQFGK-----CQERKRGRQTYTRYQTLELEKEFHFNRYLTR 324
HXA1_HUMAN  ETSSPAQTFDWMKVKRNP PKTGKVG EYGYLGQPNVARTNFTTKQLTELEKEFHFNKYLTR 256
          : :  * :  :  ** :  :  :  :  *  :  *  *  ***** :  *****

ANTP_DROME  RRRIEIAHALCLTERQIKIWFQNR RMKWKKENKTKGEPGSG-----GEGDEITP---- 373
HXA1_HUMAN  ARRVEIAASLQLNETQVKIWFQNR RMKQKKREKEGLLPISPATPPGNDEKAEESSEKSSS 316
          ** : ***  : *  *  *  * : ***** ** : : *  *  : : : *  :

ANTP_DROME  -PNSPQ----- 378
HXA1_HUMAN  SPCVPSPGSSTSDTLTTS H 335
          *  *
    
```

# Локальное выравнивание белков



```
ANTP_DROME   QFGKQCERKRGRQTYTRYQTLELEKEFHFNRYLTRRRRIEIAHALCLTER 340
HXA1_HUMAN   EGYLGQPNAVRTNFTTKQLTELEKEFHFNKYLTRARRVEIAASLQLNET 275
              ..*      .      *      *      *****      .      **      *      *      *
              .

ANTP_DROME   QIKIWFQNRRMKWKKENK 358
HXA1_HUMAN   QVKIWFQNRRMKQKKREK 289
              *      *****      **      *
```

Программа **глобального** выравнивания только расставляет гэпы = выравнивает последовательности по всей длине

Программа **локального** выравнивания:

- 1) выбирает в каждой последовательности по участку;
- 2) выравнивает между собой эти участки.





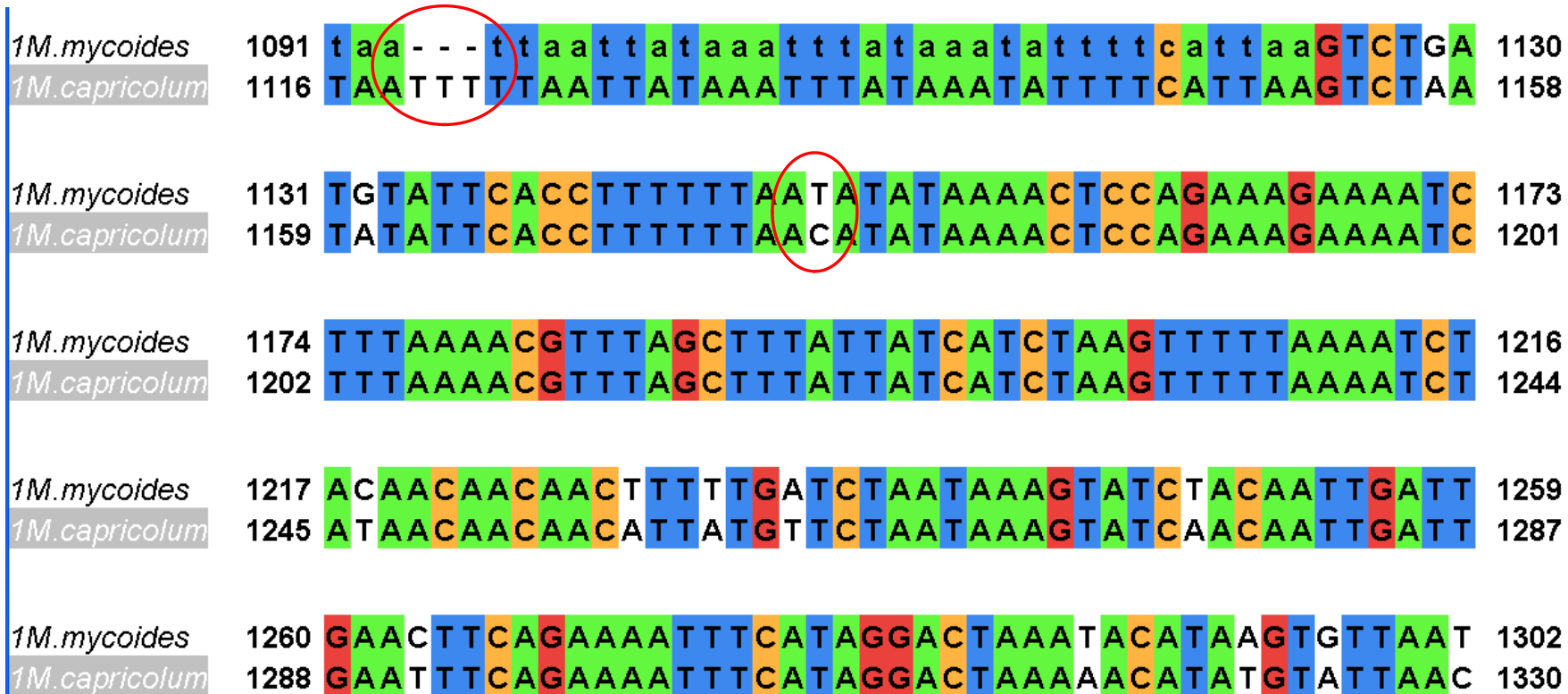
## Когда какое выравнивание нужно

- Если мы уверены, что две последовательности гомологичны по всей длине, то глобальное
- Если две последовательности содержат (относительно небольшие) гомологичные участки, то локальное
- Если мы ничего заранее не знаем, то предпочтительно тоже локальное (на глобальном можно не увидеть хороший участок сходства — признак гомологии)

# Напоминание

- Гомология – происхождение от общего предка
- Выравнивание последовательностей должно отражать эволюцию
- Выравнивание имеет биологический смысл только для гомологичных участков геномов или белков

# Участок выравнивания двух геномов





# Алгоритмы выравнивания

- Парное глобальное выравнивание (global alignment of two sequences) — алгоритм Нидлмана – Вунша (Needleman & Wunsch)
- Парное локальное выравнивание (local alignment of two sequences) — алгоритм Смита – Уотермена (Smith & Waterman)
- Множественное выравнивание (multiple alignment), когда последовательностей больше двух — великое множество алгоритмов, самые известные: ClustalW, Muscle, MAFFT, T-Coffee.



# Принцип работы алгоритмов парного выравнивания

- Имеется процедура вычисления **веса** выравнивания (alignment score)
- Алгоритм Нидлмана – Вунша находит оптимальную расстановку гэпов: такую, чтобы вес получившегося выравнивания был **максимальным**
- Алгоритм Смита – Уотермена находит оптимальные (то есть дающие максимальный вес):
  - пару участков в каждой последовательности;
  - для этой пары — расстановку гэпов.



# Как вычисляется вес выравнивания

- Цель: вес должен быть тем больше, чем больше выравнивание похоже на правильное (отражающее эволюцию). В настоящее время устоялся следующий подход:
- Для последовательностей ДНК за каждую пару совпадающих букв к весу выравнивания **прибавляется** некоторое число (например, 1), а за каждую пару разных букв **вычитается** другое число (например, 2)
- Для последовательностей белков есть понятие веса замены букв: чем реже буквы заменяются друг на друга в эволюции, тем меньше этот вес  
Всего нужно задать 210 чисел, некоторые из них положительные, другие отрицательные. Они составляют **матрицу весов аминокислотных замен**.
- За гэпы вычитается **штраф** (gap penalty)

# Матрица весов аминокислотных замен BLOSUM 62

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	9																				
S	-1	4																			
T	-1	1	5																		
P	-3	-1	-1	7																	
A	0	1	0	-1	4																
G	-3	0	-2	-2	0	6															
N	-3	1	0	-2	-2	0	6														
D	-3	0	-1	-1	-2	-1	1	6													
E	-4	0	-1	-1	-1	-2	0	2	5												
Q	-3	0	-1	-1	-1	-2	0	0	2	5											
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8										
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5									
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5								
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5							
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4						
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4					
V	-1	-2	0	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4				
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6			
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7		
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11	

Треугольная (симметричная) матрица

# Замечания

- Программы выравнивания действуют формально и выдадут выравнивание, даже если на вход им подать негомологичные белки. Смысла такое выравнивание иметь не будет
- Даже если белки гомологичны, в выравнивании, выданном программой, не обязательно всюду будут сопоставлены гомологичные аминокислотные остатки или нуклеотиды. Алгоритмы выравнивания сделаны так, чтобы максимизировать долю правильных сопоставлений. Но невозможно придумать такой алгоритм, который бы выдавал биологически правильный ответ всегда. Выравнивание, выданное программой — это всего лишь реконструкция причины (эволюции) по последствиям (т. е. современным последовательностям).





# Вопросы и ответы

## Что такое гомология?

Ответ: общность происхождения

*(НЕПРАВИЛЬНО говорить «последовательности гомологичны на 56%.  
Последовательности либо гомологичны, либо нет)*

## Как определить, гомологичны ли два белка?

Ответ: в большинстве случаев единственный способ — выровнять их последовательности и посмотреть на процент совпадающих букв.

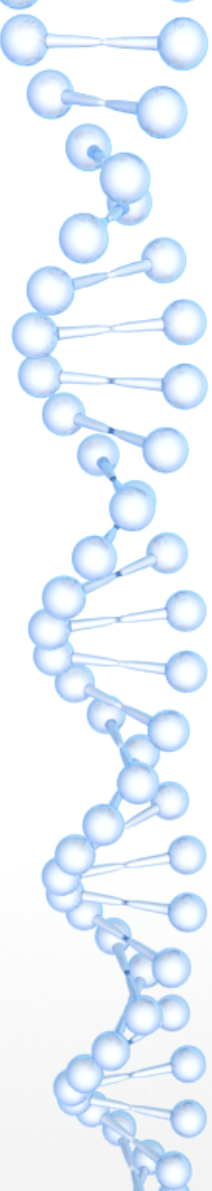
Если он достаточно велик, то белки, вероятно, гомологичны.

Если нет, то всякое может быть.

Если для обоих белков известны пространственные структуры, то есть гораздо более чувствительный способ: сравнить ход полипептидной цепи.

## Какой процент идентичности служит надёжным признаком гомологии?

Ответ: для белков обычно более 20–25% на достаточно длинном участке (а точнее см. дальше про E-value)



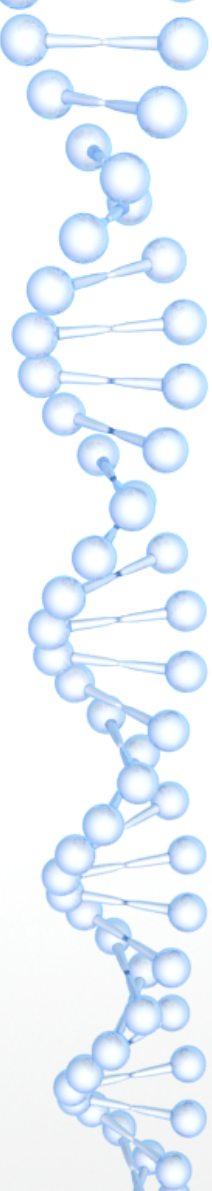
**BLAST – программа поиска  
последовательностей, похожих  
на данную**

**“Basic Local Alignment Search Tool”**



# Задача поиска

- Последовательность белка несет мало информации
- Больше информации можно получить из сравнения последовательностей
  - определение гомологии белков, предсказание функции, реконструкция филогении, ...
- Для сравнения последовательности надо выровнять
- О гомологичности судят по степени сходства последовательностей
- Численное выражение степени сходства — вес выравнивания
- Критерий гомологичности: сходство такое, какое не могло бы быть получено случайно
- Важная задача: поиск гомологичных последовательностей среди всех известных



# Формулировка задачи поиска по СХОДСТВУ

- **Дано:**
  - последовательность белка;
  - банк последовательностей, например, Uniprot или часть его, состоящая из всех белков бактерий
- **Требуется:** получить список белков, гомологичных данному белку (полностью или частично)
- **Решение:** список последовательностей со сходством больше порога
  - порог должен отражать степень неслучайности



# Решение задачи поиска по сходству

- Для каждой *банковской последовательности* строим выравнивание с *данной* последовательностью
  - По выравниванию решаем, гомологичны ли белки
  - Если да, то дописываем в список находок
- Проблема: локальное или глобальное выравнивание?
- Ещё две проблемы:
  - справится ли компьютер
  - как принять решение о гомологичности



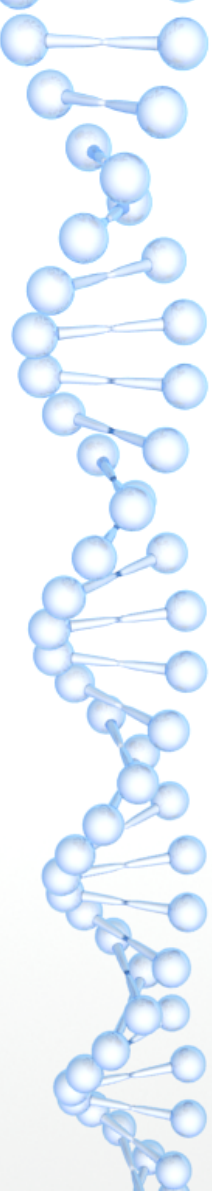
# Решение задачи поиска по сходству

- Для каждой *банковской последовательности* строим **локальное** выравнивание с *данной* последовательностью
  - По выравниванию решаем, гомологичны ли белки
  - Если да, то дописываем в список находок
- Применять алгоритм Смита – Уотермена оказывается неудобно: он хорош для двух последовательностей, но миллион выравниваний будет строить слишком долго
- Выход: придумать **быстрый эвристический алгоритм**



# Точные и эвристические алгоритмы

- Точный алгоритм решает точную задачу: формализацию содержательной задачи
  - Пример: алгоритм Смита – Уотермена решает точную задачу: всегда находит выравнивание с наибольшим весом (для некоторого способа вычисления веса)
- Для эвристического алгоритма точную задачу сформулировать нельзя, но он тем не менее выдаёт что-то, что (если алгоритм хороший) достаточно часто приближает нас к решению содержательной задачи
  - Примеры: алгоритм BLAST, все алгоритмы множественного выравнивания.



# BLAST: быстрый эвристический алгоритм поиска сходных последовательностей

- BLAST сначала отбирает те последовательности и места (порядковые номера букв) в них, с которых имеет смысл начать строить выравнивание

*"The central idea of the BLAST algorithm is that a statistically significant alignment is likely to contain a high-scoring pair of aligned words."*

*S.F. Altschul et al., NAR 1997*

- Для этого **индексируются** все слова небольшой длины ( $W = 6$  по умолчанию) во всех последовательностях банка



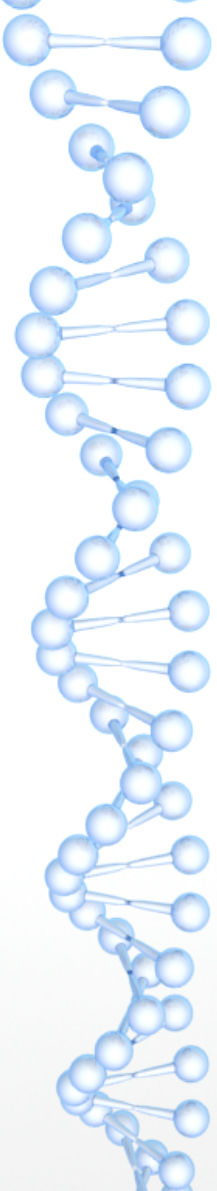


Индекс — примерно то же, что алфавитный указатель в книге

## АЛФАВИТНЫЙ УКАЗАТЕЛЬ

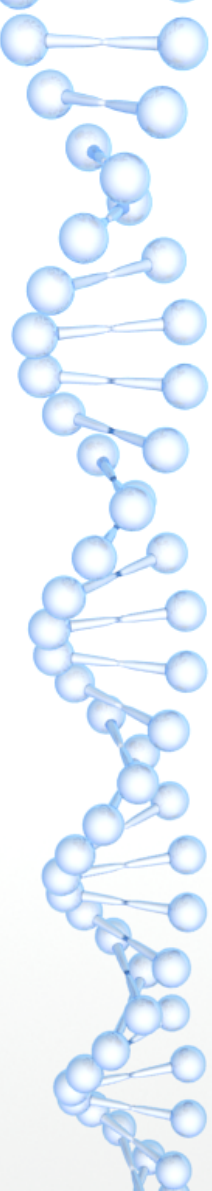
*(цифры обозначают номера экспериментов или параграфов)*

- |   |   |
|---|---|
| Агрегатное состояние 18, 19.                      | Время, деление на равные промежутки 15, 16. |
| Акустический указатель 169.                       | Время, измерение 13—15, 113, § 3.           |
| Акция 128.  | Время падения 120.                          |
| Амплитуда колебания 162, 191, 196, 197, 211, 217. | Высота падения 118, 120.                    |
| Аперiodические колебания 205.                     | Вытесняемость жидкости 8, 9, 21, 22.        |
| Балансирование 65, 66, 70.                        | Вытесняемость твердых тел 20.               |
| Барометр чашечный § 1.                            | Гармоническое колебание 191, 196, § 28.     |
| Батавские слезки 61.                              | Градуирование шкалы динамометра 55.         |
| Биение 217.                                       | Грамм § 7.                                  |
| Бифилярный подвес 150, 156, 162, 197, 207.        | Графики 55, 147, 183, 193, 194, 199.        |
| Блок 84—86, § 2 — 1, 3, 4.                        | Грузики с крючками § 2—10.                  |
| Блок ступенчатый § 2—5.                           | Давления, сила 53, 135.                     |
| Болонская колбочка 61.                            | Дальность полета 118, 122, 157.             |
|   | Движение волновое 201.                      |



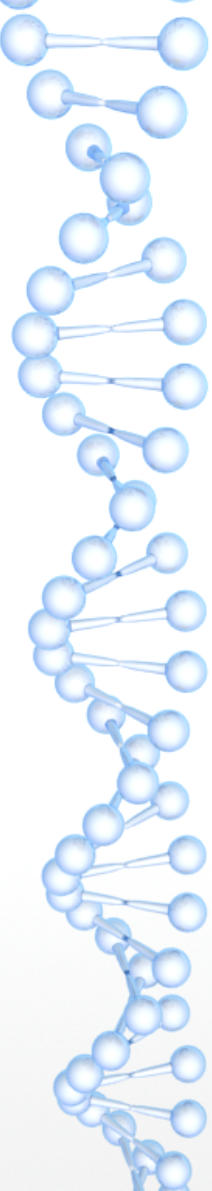
# Для BLAST по банку последовательностей ГОТОВИТСЯ ТАБЛИЦА

- |          |   |
|----------|---|
| • Слово  | • Где встречается   |
| • AMNNRA | • PPP_ECOLI 51, 237; QQQ_HUMAN<br>976; SSS_DROME 17, 111; ..... |
| • .....  | • .....   |
| • FALTGG | • .....   |



# При поиске прежде всего создаётся список всех слов, похожих на слова входной последовательности

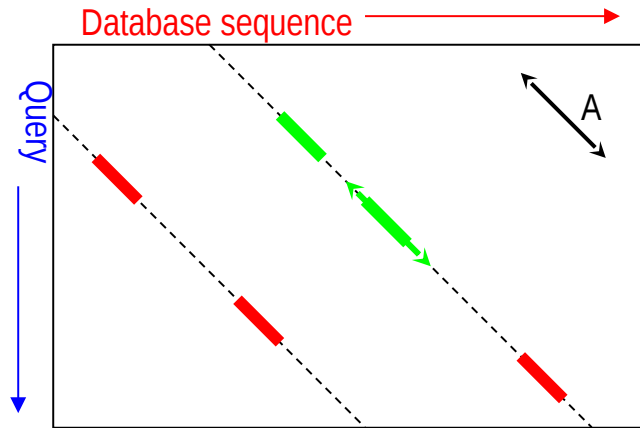
- Входная последовательность (query): **QLGVKAGW**
  - пусть длина слова  $W = 3$  (это параметр программы)
  - пусть два слова считаются похожими, если вес выравнивания слов больше или равен  $T = 13$  (это тоже параметр программы)
- Шаг 1: запомним все слова длины  $W$  во входной последовательности: **QLG LGV GVK VKA KAG AGW**
- Шаг 2: Расширим список, добавив похожие слова:
  - например, для GVK это **GAK GTK G GK GLK**



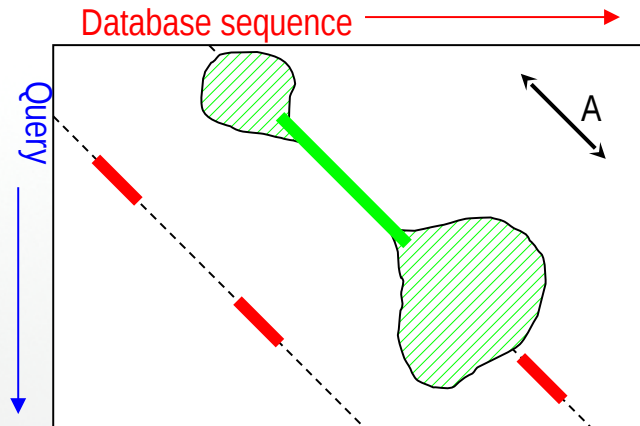
## Отбор банковских последовательностей для выравнивания

- Шаг 3: По индексной таблице составляем список всех слов в банке, похожих на слова входной последовательности
  - Для отбора последовательности необходимы ДВА слова на расстоянии  $A$  ( $A$  — это тоже параметр, по умолчанию  $A = 20$ ) причем на ОДНОЙ ДИАГОНАЛИ (след. слайд)

# Отбор последовательности: два слова на одной диагонали на расстоянии меньше $A$ (зеленые)



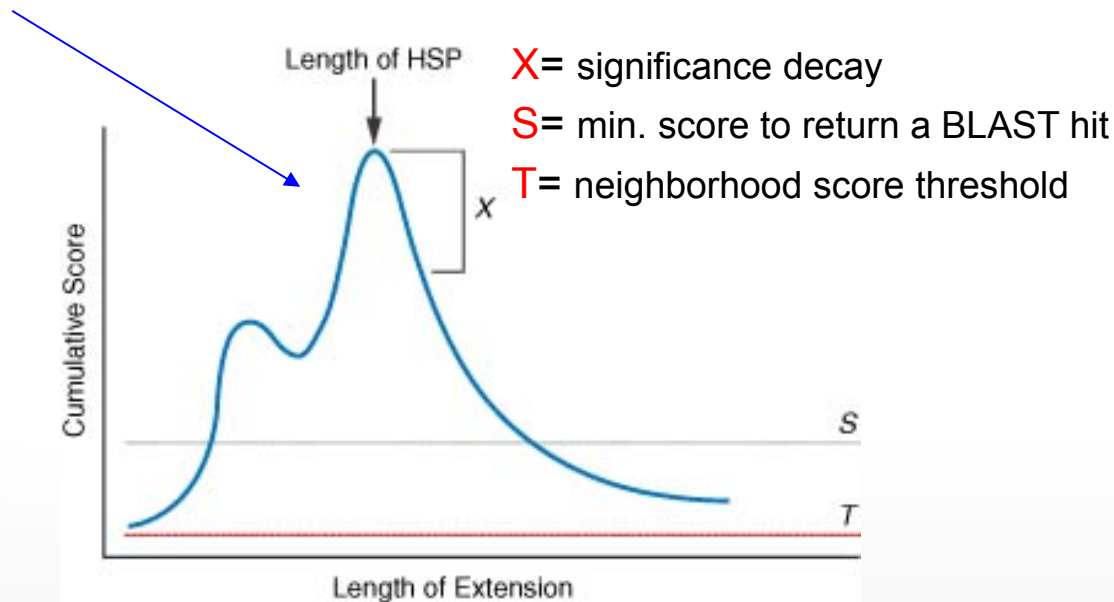
Ungapped extension if:  
2 "Hits" are on the same diagonal but at  
a distance less than  $A$



Extension using [dynamic programming](#)  
limited to a restricted region  
limited through a [score drop-off](#) threshold

# Выравнивание начинается с найденных слов

- Выравнивание расширяется, начиная с найденных слов, в обе стороны
- Критерий остановки см. на рисунке





# Роль длины слова. Мой эксперимент

- Вход: последовательность из 466 остатков
- NCBI BLAST (<https://blast.ncbi.nlm.nih.gov/>)
- Область поиска: Swissprot, белки из бактерий
- Параметры, кроме "Word Size", по умолчанию. В частности, порог E-value = 10
- **W = 6**
  - Найдено 16 последовательностей, в них 18 выравниваний
  - 8 находок с  $E < 0.001$
  - Время работы сервиса NCBI – менее одной минуты
- **W = 2**
  - Найдено 69 последовательностей, в них 75 выравниваний
  - 12 находок с  $E < 0.001$
  - Время работы сервиса NCBI – около 35 мин



# Программа BLAST. Оценка находок

- Результатом работы BLAST является список находок
- Каждая находка представляет из себя локальное выравнивание входной последовательности и банковской
- Выравнивание имеет вес, чем больше вес – тем лучше находка.
- Есть много матриц весов; можно менять и штрафы за гэпы

Как оценить, может ли выравнивание с данным весом быть получено случайно?





# Программа BLAST. Оценка находок

- Для каждой находки BLAST вычисляет т.н. E-value
- E-value это математическое ожидание числа находок с таким же или большим весом выравнивания в банке случайных последовательностей того же размера, как тот банк, в котором велся поиск  
(или, эквивалентно, в том же банке, но для случайной входной последовательности той же длины и состава)
- E-value зависит от объёма банка, в котором ведётся поиск. Чем больше банк, тем больше шансов найти в нем выравнивание с данным или большим весом
- Маленькое значение E-value можно интерпретировать как (маленькую) вероятность того, что находка ошибочная  
(то есть E-value = 0,001 ≈ вероятность ошибки одна тысячная)
- Карлин и Альтшуль ([Karlin & Altschul, PNAS, 1990](#)) решили математическую задачу, позволяющую рассчитать E-value по весу, длине последовательности и объёму банка, не проводя каждый раз эксперименты с поиском в случайном банке.
- Формула Карлина и Альтшуля: **E-value =  $L_{\text{query}} \cdot L_{\text{bank}} \cdot K \cdot \text{EXP}(-\text{Score} \cdot \lambda)$**   
**Score** — это вес выравнивания, а параметры  $K$  и  $\lambda$  зависят от матрицы замен и штрафов за гэпы (их получили один раз для каждой матрицы и параметров штрафов путём вычислительного эксперимента)



# Нормализованный вес (в битах)

$$B = \text{Bit score} = (\lambda \cdot \text{Score} - \ln K) / \ln 2$$

$$\text{Тогда E-value} = L_{\text{query}} \cdot L_{\text{bank}} / 2^B$$

**Нормализованный вес для выравниваний без гэпов не зависит от матрицы весов и штрафов**

Увеличение веса на 1 бит означает «уменьшение неопределённости вдвое», то есть вдвое меньшую вероятность получить выравнивание с таким же весом по случайным причинам.



# Задачи, которые решают с помощью BLAST

- Есть ли моя последовательность в банке?
- Есть ли у моего белка аннотированные гомологи? А гомологи с известной пространственной структурой?
- Хочу гомологи моего белка, чтобы:
  - предсказать его функцию *или*
  - понимать, какие его части консервативны  $\Rightarrow$  важны для функции *или*
  - мало ли ещё для чего...
- Закодирован ли где-нибудь в данном новом геноме белок с интересующей меня функцией?
- Я нашёл в геноме участок, похожий на кодирующий. Есть ли в банке белки, похожие на него?

# Интерфейс на сайте NCBI

<http://blast.ncbi.nlm.nih.gov/> → Protein blast

BLAST Basic Local Alignment Search Tool

Home Recent Results Saved Strategies Help

NCBI/ BLAST/ blastp suite

blastn blastp blastx tblastn tblastx

BLASTP programs search protein databases using a protein query. [more...](#)

Enter Query Sequence

Enter accession number(s), gi(s), or FASTA sequence(s)  Clear Query subrange

From  To

Or, upload file  Файл не выбран

Job Title

Enter a descriptive title for your BLAST search

Align two or more sequences

Choose Search Set

nr	Non-redundant Reference Sequences
refseq	Reference Sequences
swissprot	SWISS-PROT
pat	Patents
pdb	Protein Data Bank
env_nr	Environmental samples

Database

Organism Optional

Exclude  Exclude

Exclude Optional  Models (XM/XP)  Uncultured/environmental sample sequences

Entrez Query Optional

Program Selection

Algorithm  blastp (protein-protein BLAST)

PSI-BLAST (Position-Specific Iterated BLAST)

PHI-BLAST (Pattern Hit Initiated BLAST)

Choose a BLAST algorithm

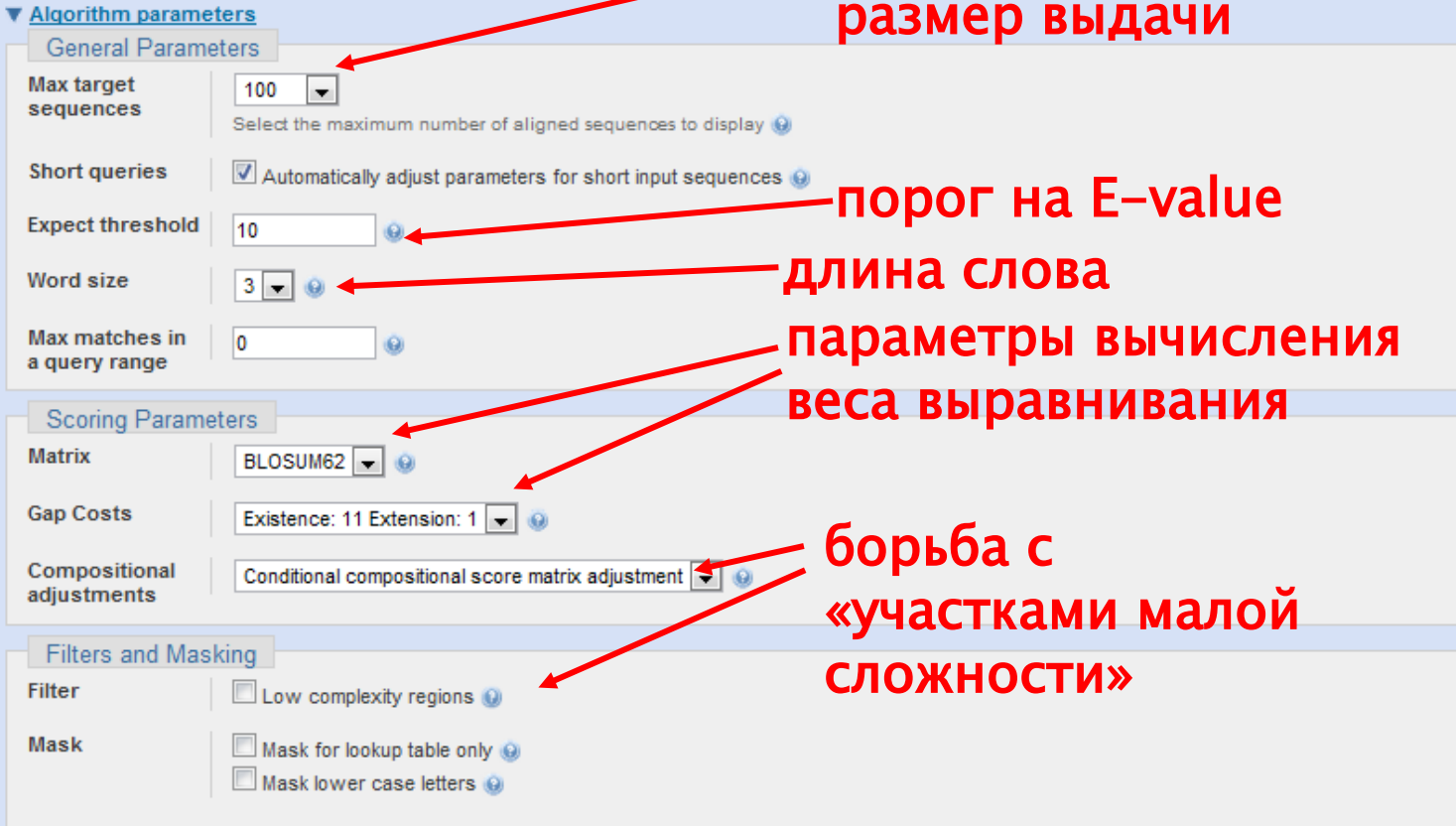
**BLAST** Search database Non-redundant protein sequences (nr) using Blastp (protein-protein BLAST)

Show results in a new window

[Algorithm parameters](#)

# Интерфейс на сайте NCBI

## Дополнительные параметры



**максимальный размер выдачи**

**порог на E-value**

**длина слова**

**параметры вычисления веса выравнивания**

**борьба с «участками малой сложности»**

**Algorithm parameters**

**General Parameters**

Max target sequences: 100  
Select the maximum number of aligned sequences to display

Short queries:  Automatically adjust parameters for short input sequences

Expect threshold: 10

Word size: 3

Max matches in a query range: 0

**Scoring Parameters**

Matrix: BLOSUM62

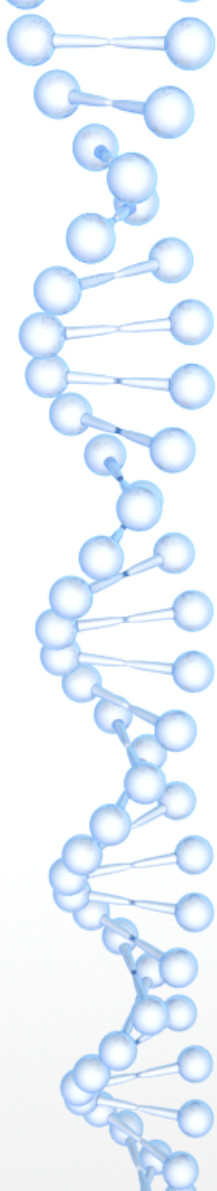
Gap Costs: Existence: 11 Extension: 1

Compositional adjustments: Conditional compositional score matrix adjustment

**Filters and Masking**

Filter:  Low complexity regions

Mask:  Mask for lookup table only  
 Mask lower case letters



## Пример участка малой сложности: GAR (глицин-аргинин богатый) домен

XM_004710492.2_Echinops_telfairi	16	GRGGGF	FGDRGGF	GGRGGF	GGDRGG	-	RGGRGGF	46
XM_004313699.1_Tursiops_truncatus	45	PRGGGF	SGRGGF	GDRGGR	GGGRGGF	GGGRGGF	GGF	76
XM_004620749.2_Sorex_araneus	7	PRGGGF	GGRGGF	GDRGGR	GGG	-	-	34
XM_004599705.2_Ochotona_princeps	7	PRGGGF	GGRGGF	GDRGGR	GGR	-	-	34
XM_012546826.1_Sarcophilus_harrisii	24	PRGGG	-	-	RGGR	GR	GRGGDRGG	50
XM_004710492.2_Echinops_telfairi	47	GGGRGR	-	GG	S	-	-	71
XM_004313699.1_Tursiops_truncatus	77	GGGR	-	-	GGFGEGRGR	GGGFRGRG	-	104
XM_004620749.2_Sorex_araneus	35	-	GGRGR	GGGFR	GGRGR	GGGGRGGG	-	64
XM_004599705.2_Ochotona_princeps	35	GGGRGR	GGGFRG	-	GGGGGGRGGG	-	GGGGGRG	64
XM_012546826.1_Sarcophilus_harrisii	51	-	GGRGR	-	GGF	GA	-	78
XM_004710492.2_Echinops_telfairi	72	GGFQP	-	GGT	RGRGRGGR	RGGNQSGK		96
XM_004313699.1_Tursiops_truncatus	105	GGFQS	-	GGSRGRGRGGR	-	KNQSGK		127
XM_004620749.2_Sorex_araneus	65	GGFQP	GGGN	RGRGRGK	-	RGGQSGK		88
XM_004599705.2_Ochotona_princeps	65	GGFHS	-	GGNRGRGRGK	-	KNQSGK		87
XM_012546826.1_Sarcophilus_harrisii	79	R	-	-	-	GT	PRGRGRGGR	97



# Что выдаёт BLAST

- **Список последовательностей, предположительно гомологичных «запросу» (query)**  
для каждой находки приведены: E-value (“Expect”), вес в битах (Score), процент идентичности выравнивания и процент покрытия запроса выравниванием.
- **Локальные выравнивания запроса с каждой из находок**  
рядом с каждым выравниванием приведены некоторые его характеристики: проценты идентичности, сходства (“Positives”), гэпов, краткое описание находки, вес: обычный и в битах, опять-таки E-value,





# Выдача BLAST

## (фрагмент — верхняя часть списка находок)

RID: B54B34KT014

Job Title:P00174:RecName: Full=Cytochrome b5

Program: BLASTP

Query: RecName: Full=Cytochrome b5 [Gallus gallus] ID: P00174.4(amino acid) Length: 138

Database: swissprot Non-redundant UniProtKB/SwissProt sequences

Sequences producing significant alignments:

Description	Max Score	Total Score	Query cover	E Value	Per. Ident	Accession
RecName: Full=Cytochrome b5 [Gallus gallus]	286	286	100%	1e-100	100.00	P00174.4
RecName: Full=Cytochrome b5 [Oryctolagus cuniculus]	220	220	89%	3e-74	79.84	P00169.4
RecName: Full=Cytochrome b5; AltName: Full=Microsomal cytochro...	219	219	89%	3e-74	79.03	P00167.2
RecName: Full=Cytochrome b5 [Sus scrofa]	219	219	89%	4e-74	79.03	P00172.3
RecName: Full=Cytochrome b5 [Rattus norvegicus]	219	219	89%	5e-74	78.23	P00173.2
RecName: Full=Cytochrome b5 [Mus musculus]	216	216	89%	8e-73	77.42	P56395.2
RecName: Full=Cytochrome b5 [Equus caballus]	215	215	89%	2e-72	76.61	P00170.3
RecName: Full=Cytochrome b5 [Bos taurus]	208	208	89%	9e-70	74.19	P00171.3
RecName: Full=Cytochrome b5 [Alouatta seniculus]	154	154	59%	7e-49	81.71	P00168.2
RecName: Full=Cytochrome b5 type B; AltName: Full=Cytochrome b...	142	142	89%	4e-43	51.20	P04166.2
RecName: Full=Cytochrome b5 type B; AltName: Full=Cytochrome b...	140	140	89%	2e-42	50.81	Q9CQX2.1
RecName: Full=Cytochrome b5 type B; AltName: Full=Cytochrome b...	138	138	99%	9e-42	44.83	O43169.3
RecName: Full=Cytochrome b5 type B; AltName: Full=Cytochrome b...	135	135	99%	1e-40	43.45	Q5RDJ5.3
RecName: Full=Cytochrome b5; Short=CYTB5 [Drosophila...	126	126	88%	3e-37	47.15	Q9V4N3.1
RecName: Full=Cytochrome b5; Short=CYTB5 [Musca domestica]	119	119	88%	2e-34	43.09	P49096.1
RecName: Full=Cytochrome b5 [Rhizopus stolonifer]	99.0	99.0	86%	2e-26	39.68	Q9HFV1.1
RecName: Full=Cytochrome B5 isoform D; Short=AtCb5-D; AltName:...	97.8	97.8	89%	7e-26	36.22	Q9ZWT2.1
RecName: Full=Cytochrome b5 [Nicotiana tabacum]	96.7	96.7	59%	2e-25	47.56	P4408.1
RecName: Full=Cytochrome b5 [Borago officinalis]	95.9	95.9	65%	4e-25	41.49	O04354.1
RecName: Full=Cytochrome b5 [Mortierella alpina]	94.7	94.7	83%	1e-24	37.61	Q9Y706.1



# Выравнивание, выданное BLAST

Length=129      Number of matches=1

Длина найденного белка

Вес в битах      Вес      E-value

Score = 78.6 bits (192), Expect = 9e-15, Method: Compositional matrix adjust.  
Identities = 34/73 (47%), Positives = 50/73 (68%), Gaps = 0/73 (0%)

Query 17 YRLEEYQKHNN SQSTWIIIVHHRIYDITKFLDEHPGGEEVLREQAGGDATENFEDVGHSTD 76  
          Y EEV +H            W+I++ ++Y+I+ ++DEHPGGEEV+ + AG DATE F+D+GHS +  
Sbjct 11 YTHEEVAQHTTHDDLWVILNGKVYNISNYIDEHPGGEEVILDCAGTDATEAFDDIGHSDE 70

Query 77 ARALSETFIIGEL 89  
          A + E IG L  
Sbjct 71 AHEILEKLYIGNL 83

Число совпадений      Число сходных "букв"      Число символов гэпа  
Длина выравнивания

**Выравнивание локальное!** В данном случае участок 17–89 запроса выровнен с участком 11–83 находки (а вся находка длиной 129 — заметно больше!).



# Поиск в нуклеотидных банках

- Всё предыдущее относилось в основном к программе BLASTP из пакета BLAST, предназначенной для поиска гомологов белков в банке белковых последовательностей
- Есть прямой аналог для нуклеотидных последовательностей (ДНК и РНК), называемый BLASTN. Но:
  - Поиск по гомологии для ДНК/РНК вообще гораздо менее надёжен, чем для белков (потому что выравнивания получаются хуже: из четырёх букв чаще возникают случайные совпадения, и нет таких информативных матриц замен, как для белков).
  - К тому же, чтобы получить реальное ускорение, длина слова в BLASTN была вначале поставлена равной 11, что отсекало значительную часть даже тех гомологов, выравнивания которых достоверно отличались от случайных
- Поэтому программа BLASTN была разделена на две: Megablast с длиной слова 28 для быстрого поиска последовательности по фрагменту и собственно BLASTN. Последним, если поставить длину слова поменьше (7 хотя бы), можно искать гомологи не кодирующих фрагментов ДНК и РНК.



## ... и ещё три разновидности BLAST

- Для поиска кодирующих последовательностей в **нуклеотидных** банках используется программа **TBLASTN**, которая формально транслирует каждую банковскую последовательность «в шести рамках» (то есть переводит её в шесть аминокислотных последовательностей).  
*(Понятно, почему шесть? Если нет, подумайте...)*  
Запросом служит белок, а целью — найти в банке последовательности, кодирующие его гомологи.
- Для поиска кодирующих участков в данной последовательности ДНК используется программа **BLASTX**. Она ищет в **белковом** банке белки, похожие на то, что закодировано в запросе, запрос — последовательность ДНК.
- Наконец, создана (но редко используется) программа TBLASTX, для которой и запрос — нуклеотидная последовательность, и банк нуклеотидный. Она ищет в банке участки, кодирующие что-то похожее на то, что закодировано в запросе, и делает это, выполняя  $6 \times 6 = 36$  сравнений для каждой банковской последовательности...

Больше подробностей — на сайте NCBI:

[https://blast.ncbi.nlm.nih.gov/Blast.cgi?CMD=Web&PAGE\\_TYPE=BlastDocs](https://blast.ncbi.nlm.nih.gov/Blast.cgi?CMD=Web&PAGE_TYPE=BlastDocs)