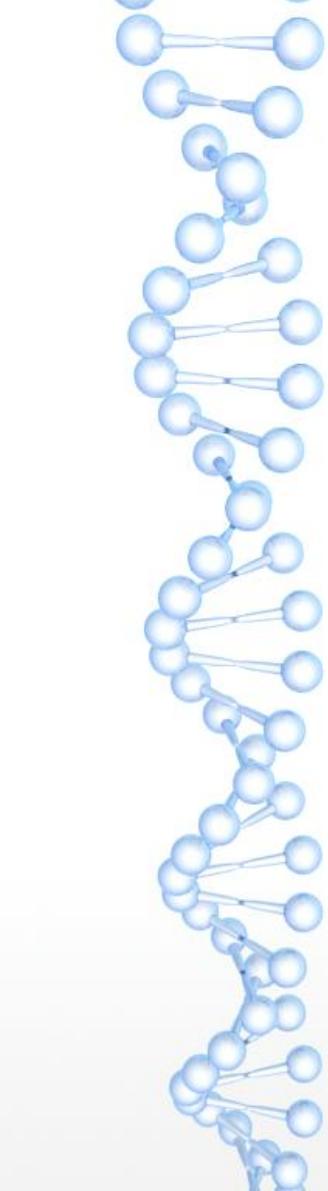


Выравнивание биологических последовательностей. Программа BLAST.

С.А. Спирин
sas@belozersky.msu.ru

МФК "Биоинформатика", 14 апреля 2021

Последовательности миоглобинов человека, мыши и быка

>MYG_HUMAN
MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDKFKHLKSEDEMKASE
DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH
PGDFGADAQGAMNKALELFRKDMASNYKELGFQG

>MYG_MOUSE
MGLSDGEWQLVLNVWGKVEADLAGHGQEVLIGLFKTHPETLDKFDKFKNLKSEEDMKGSE
DLKKHGCTVLTALGTILKKKGQHAAEIQPLAQSHATKHKIPVKYLEFISEIIIEVLKKRH
SGDFGADAQGAMSKALELFRNDIAAKYKELGFQG

>MYG_BOVIN
MGLSDGEWQLVLNAWGKVEADVAGHGQEVLIRLFTGHPETLEKFDKFKHLKTEAEMKASE
DLKKHGNTVLTALGGILKKKGHHEAEVKHLAESHANKHKIPVKYLEFISDAIIHVLHAKH
PSDFGADAQAAMSKALELFRNDMAAQYKVLGFHG

Напишем последовательности друг под другом, чтобы было лучше видно сходство

```
MYG_HUMAN MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDKFKHLKSEDEMKASE 60
MYG_MOUSE MGLSDGEWQLVLNVWGKVEADLAGHGQEVLIGLFKTHPETLDKFDKFKNLKSEEDMKGSE 60
MYG_BOVIN MGLSDGEWQLVLNAWGKVEADVAGHGQEVLIRLFTGHPETLEKFDKFKHLKTEAEMKASE 60
          **************.*******: ******** **. *****:******:**:* :**.**

MYG_HUMAN DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH 120
MYG_MOUSE DLKKHGCTVLTALGTILKKKGQHAAEIQPLAQSHATKHKIPVKYLEFISEIIIEVLKKRH 120
MYG_BOVIN DLKKHGNTVLTALGGILKKKGHHEAEVKHLAESHANKHKIPVKYLEFISDAIIHVLHAKH 120
          ****** ******* *******:* **:: **:***.**************: **.**: :*

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

Видно, что большинство букв совпадает, но некоторые различаются.
Это последовательности **гомологичных** 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
MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDKFKHLKSEDEMKASE
DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH
PGDFGADAQGAMNKALELFRKDMASNYKELGFQG

>MYG_DANRE
MADHDLVLKCWGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQGDLAGSPAVAAH
GATVLKKLGELLKAKGDHAALLKPLANTHANIHKVALNNFRLITEVLVKVMAEKAGLDAA
GQGALRRVMDAVIGDIDGYYKEIGFAG

Разная длина, как сравнивать?

Последовательности миоглобинов человека и рыбы

```
>MYG_HUMAN
MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDKFKHLKSEDEMKASE
DLKKHGATVLTALGGILKKKGHHEAEIKPLAQSHATKHKIPVKYLEFISECIIQVLQSKH
PGDFGADAQGAMNKALELFRKDMASNYKELGFQG

>MYG_DANRE
MADHDLVLKCWGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQGDLAGSPAVAAH
GATVLKKLGELLKAKGDHAALLKPLANTHANIHKVALNNFRLITEVLVKVMAEKAGLDAA
GQGALRRVMDAVIGDIDGYYKEIGFAG
```

Разная длина, как сравнивать?

Ответ: **выравнивание**

```
MYG_HUMAN MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDKFKHLKSEDEMKASE 60
MYG_DANRE ----MADHDLVLKCWGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQGD-LAGSP 55
              .: :***: ** ****  .:* *** **** :*:**: * **. :.. * : .*

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

MYG_HUMAN PGDFGADAQGAMNKALELFRKDMASNYKELGFQG 154
MYG_DANRE --GLDAAGQGALRRVMDAVIGDIDGYYKEIGFAG 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 MGLSDGEWQLVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDKFKHLKSEDEMKASE 60
MYG_DANRE ----MADHDLVLKCWGAVEADYAANGGEVLNRLFKEYPDTLKLFPKFSGISQGD-LAGSP 55
                .: :***: ** ****  .:* *** **** :*:**: * **. :.. * : .*

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

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

Выравнивание белков, гомологичных не по всей длине

```
ANTP_DROME MTMSTNNCESMTSYFTNSYMGADMHHGHYPGNGVTDLDAQQMHHYSQN----ANHQGNMP 56
HXA1_HUMAN ------------------------------------MDNARMNSFLEYPILSSGDSGTCS 24
                                                :*  :*: : :     :...*.
ANTP_DROME YPRFPPYDRMPYYN-----------------GQGMDQQQQHQVYSRPDSPSSQVGGVMPQ 99
HXA1_HUMAN ARAYPSDHRITTFQSCAVSANSCGGDDRFLVGRGVQIGSPHHHHHH-----HHHHPQPAT 79
              :*  .*:  ::                  *:*::  . *: : :      :
ANTP_DROME AQTNGQLGVPQQQQQQQQQPSQNQQQQQAQQAPQQLQQQLPQVTQQVTHPQQQQQQPVVY 159
HXA1_HUMAN YQTSGNLGVSYSHSS--CGPSYGSQNF----------------------------SAPY 108
            **.*:***  .:..    ** ..*:                               . *
ANTP_DROME ASCKLQAAVGGLGMVPEGGSPPLVDQMSGHHMNAQMTLPHHMGHPQAQLGYTD--VGVPD 217
HXA1_HUMAN SPYALNQEAD-----VSGGYPQCAPAVYSGNLSSPMVQHH-----HHHQGYAGGAVGSPQ 158
           :   *:  ..      .** *  .  : . ::.: *.  *     : : **:.  ** *:
ANTP_DROME VTE--VHQNHHNMGMYQQQSGVPPVGAPPQGMMHQGQGPPQMHQGHPGQHTPPSQNPNSQ 275
HXA1_HUMAN YIHHSYGQEHQSLALATYNNSLSP-----------------LHASHQE----ACRSP-AS 196
             .    *:*:.:.:   :..: *                  :* .*       .:.* :.
ANTP_DROME SSGMPSPLYPWMRSQFGK-----------CQERKRGRQTYTRYQTLELEKEFHFNRYLTR 324
HXA1_HUMAN ETSSPAQTFDWMKVKRNPPKTGKVGEYGYLGQPNAVRTNFTTKQLTELEKEFHFNKYLTR 256
           .:. *:  : **: : .              : :  * .:*  *  *********:****
ANTP_DROME RRRIEIAHALCLTERQIKIWFQNRRMKWKKENKTKGEPGSG-------GEGDEITP---- 373
HXA1_HUMAN ARRVEIAASLQLNETQVKIWFQNRRMKQKKREKEGLLPISPATPPGNDEKAEESSEKSSS 316
            **:*** :* *.* *:********** **.:*    * *         :.:* :
ANTP_DROME -PNSPQ------------ 378
HXA1_HUMAN SPCVPSPGSSTSDTLTTSH 335
               *  *.
```

Локальное выравнивание белков

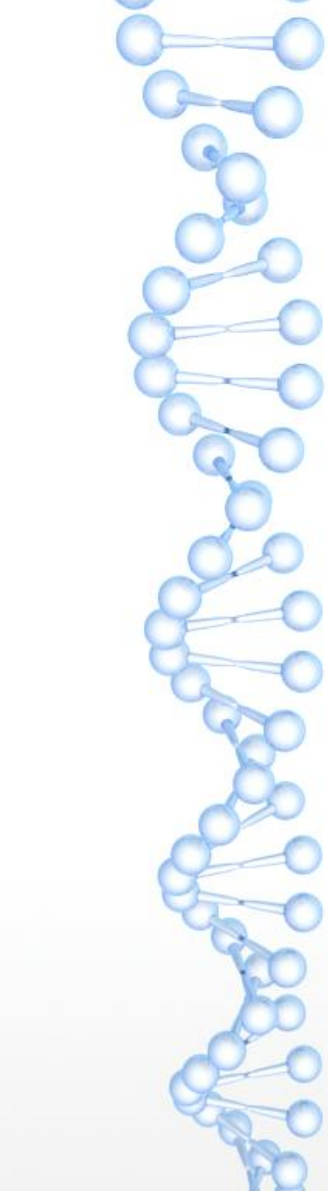
```
ANTP_DROME   QFGKCQERKRGRQTYTRYQTLELEKEFHFNRYLTRRRRIEIAHALCLTER 340
HXA1_HUMAN   EYGYLGQPNAVRTNFTTKQLTELEKEFHFNKYLTRARRVEIAASLQLNET 275
             ..*   .    *  .*  *  *********.**** **.*** .* * *

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

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

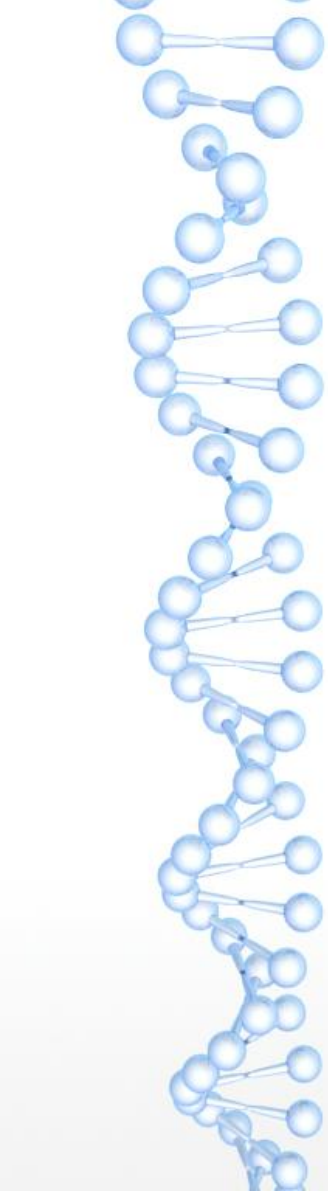
Программа **локального** выравнивания:

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



Когда какое выравнивание нужно

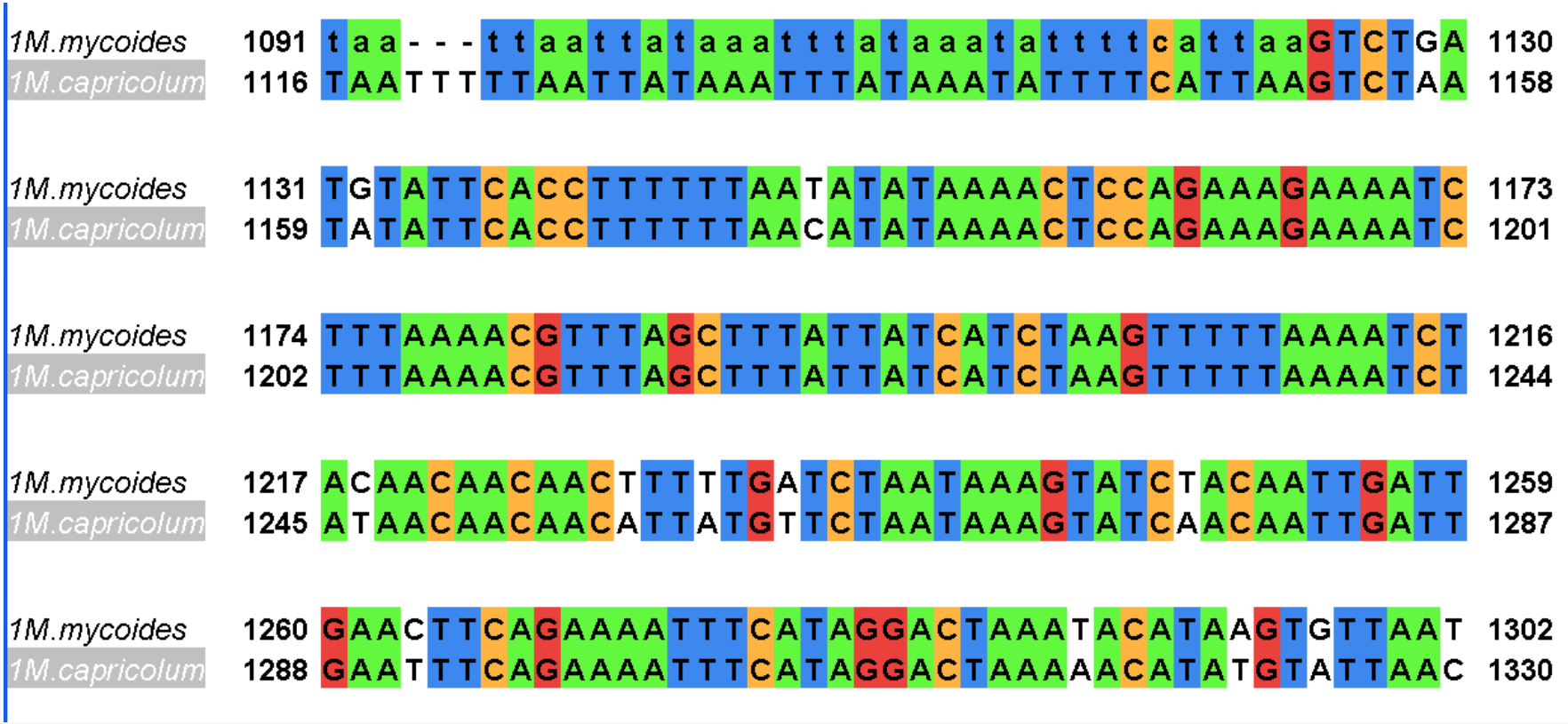
- Если мы уверены, что две последовательности гомологичны по всей длине, то глобальное
- Если две последовательности содержат (относительно небольшие) гомологичные участки, то локальное
- Если мы ничего заранее не знаем, то предпочтительно тоже локальное (на глобальном можно не увидеть хороший участок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

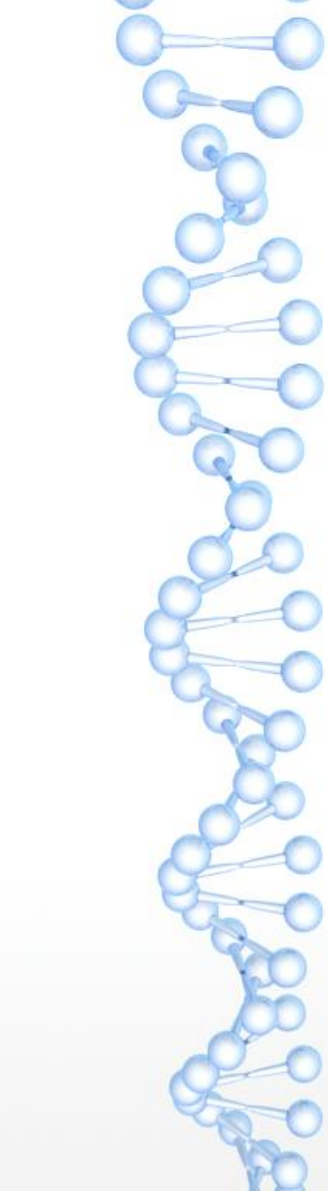


Гомология и выравнивание

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

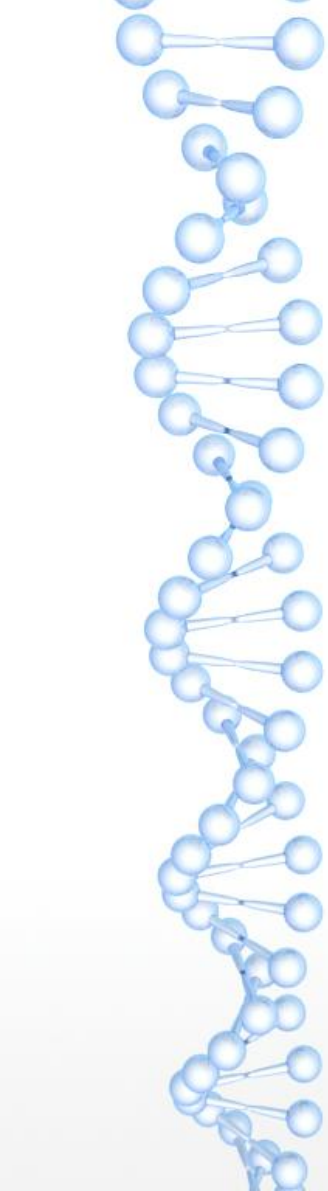
Участок выравнивания двух геномов





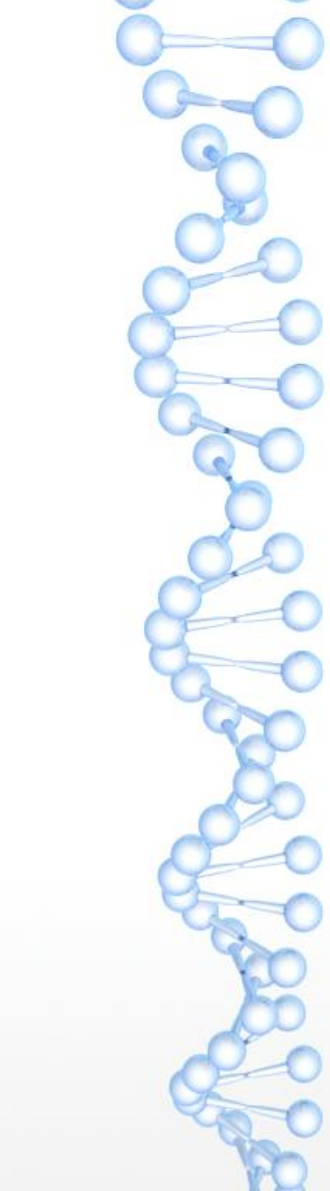
Алгоритмы выравнивания

- 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 последовательностях;
  - ✓ для этой пары — расстановку гэпов.
- Оба алгоритма используют **динамическое программирование**



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

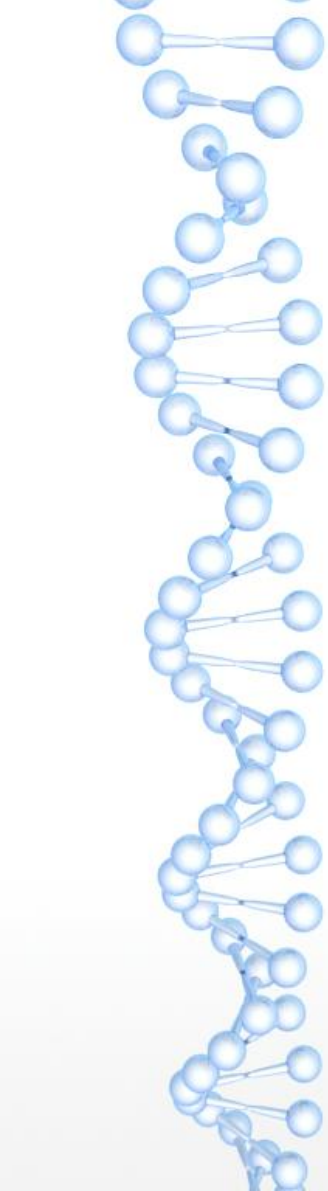
  Всего нужно задать 210 чисел, некоторые из них положительные, другие отрицательные. Они составляют **матрицу весов аминокислотных замен**.
- За гэпы вычитается **штраф** (gap penalty)

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

Треугольная (симметричная) матрица

Из работы (Henikoff&Henikoff, **1992**, *PNAS*)



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

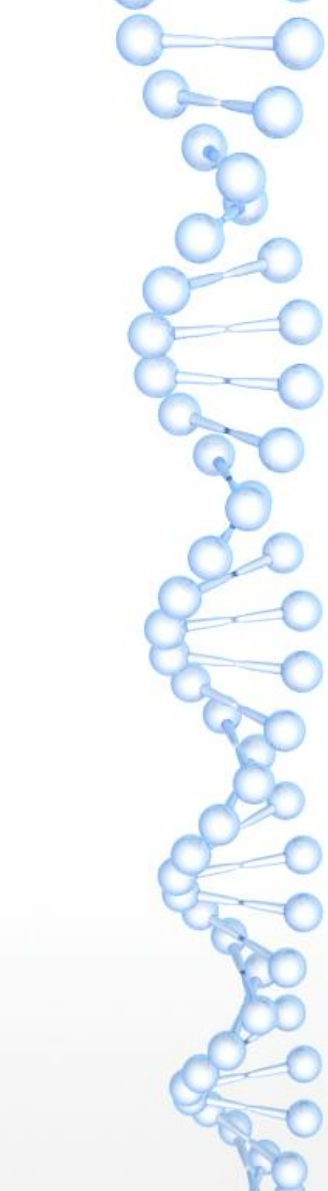
  Алгоритмы выравнивания сделаны так, чтобы максимизировать долю правильных сопоставлений. Но невозможно придумать такой алгоритм, который бы выдавал биологически правильный ответ всегда. Выравнивание, выданное программой — это всего лишь реконструкция причины (эволюции) по последствиям (т. е. современным последовательностям).

Вопросы и ответы

Что такое гомология?
Ответ: общность происхождения
(*НЕПРАВИЛЬНО говорить «последовательности гомологичны на 56%. Последовательности либо гомологичны, либо нет*)

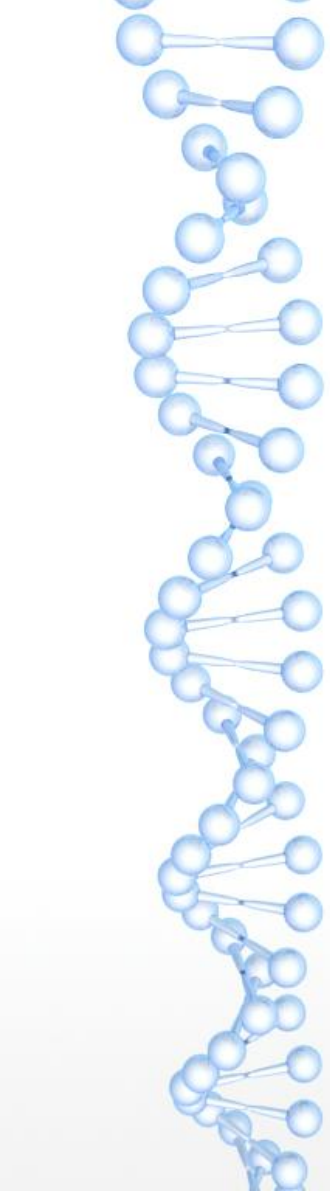
Как определить, гомологичны ли два белка?
Ответ: в большинстве случаев единственный способ — выровнять их последовательности и посмотреть на процент совпадающих букв.
Если он достаточно велик, то белки, вероятно, гомологичны.
Если нет, то всякое может быть.
Если для обоих белков известны пространственные структуры, то есть гораздо более чувствительный способ: сравнить укладку полипептидной цепи в пространстве.

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

“**B**asic **L**ocal **A**lignment **S**earch **T**ool”



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

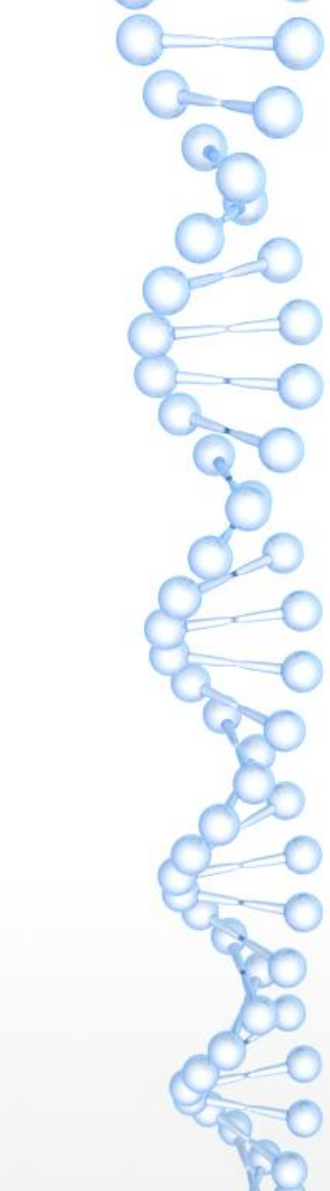
  порог должен отражать степень неслучайности

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 заданного 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

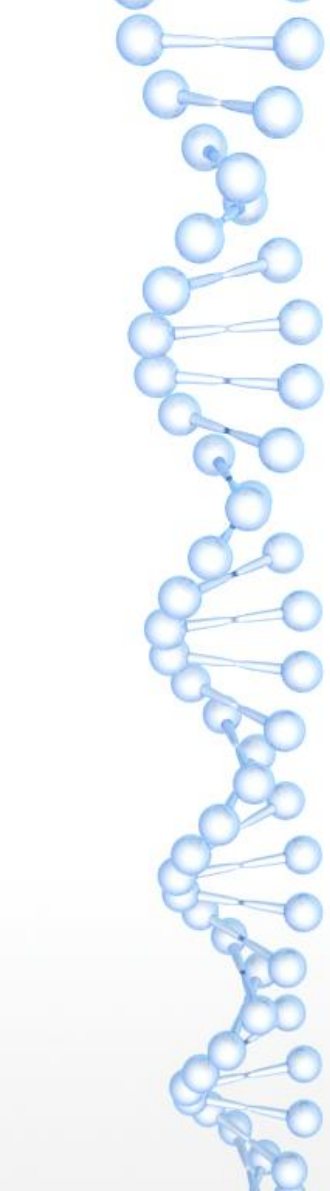
Индекс — примерно то же, что алфавитный указатель в книге

АЛФАВИТНЫЙ УКАЗАТЕЛЬ

(цифры обозначают номера экспериментов или параграфов)

По банку последовательностей готовится таблица

Слово	Где встречается
AMNNRA	PPP_ECOLI 51, 237; QQQ_HUMAN 976; SSS_DROME 17, 111; ……
……	……
FALTGG	……



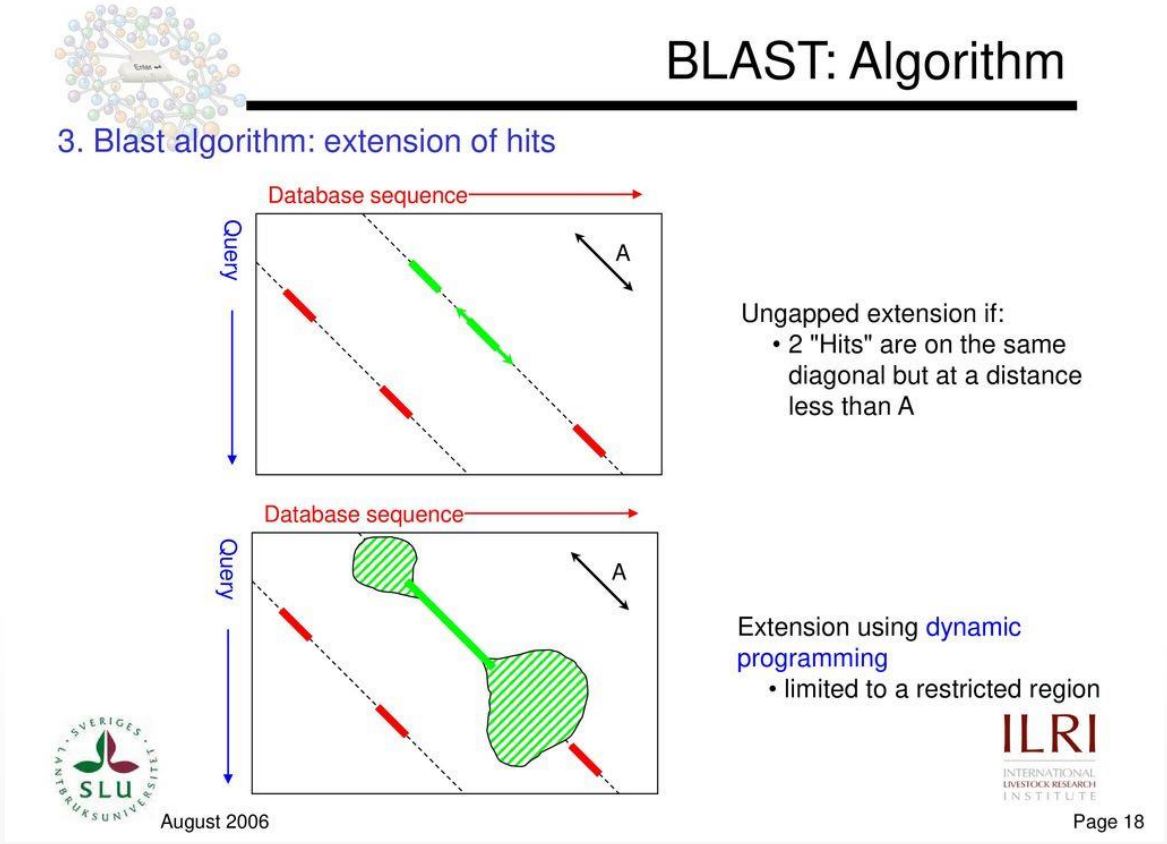
При поиске прежде всего создаётся список всех слов, похожих на слова входной последовательности

- 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 GIK GLK GVR …

Отбор банковских последовательностей для выравнивания

- Шаг 3: Используя индексную таблицу, 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** (то есть на одинаковом расстоянии во входной и банковской последовательностях)
    см. https://slideplayer.com/slide/13025199/ , слайды 16 и 18
    см. также
    http://steipe.biochemistry.utoronto.ca/abc/index.php/BLAST#Slide_0010

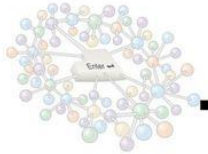
Отбор банковских последовательностей для выравнивания



https://slideplayer.com/slide/13025199/

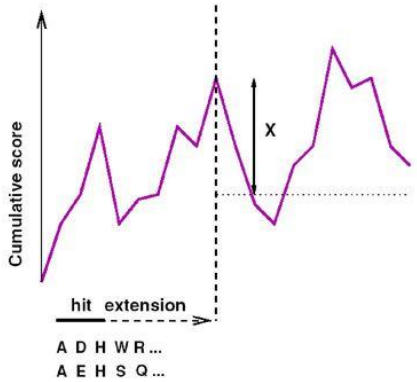
Выравнивание начинается с найденных слов

- Выравнивание расширяется, начиная с найденных слов, в обе стороны
- Критерий остановки см. на рисунке (X — тоже параметр программы)

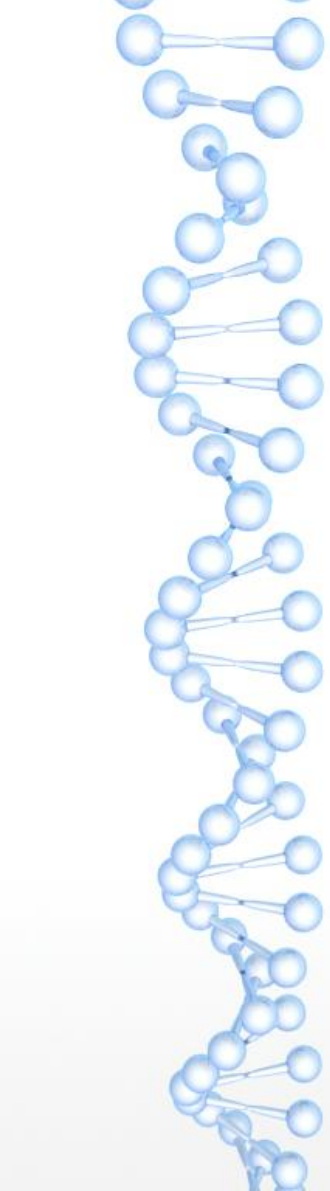


BLAST: Algorithm

Ungapped extension of hits



Each match is then extended. The extension is stopped as soon as the score decreases more then X when compared with the highest value obtained during the extension process



Достоинства и недостатки эвристического алгоритма

BLAST — эвристический алгоритм
(в отличие от точного алгоритма Смита – Уотермена)

Достоинство: скорость. Динамическое программирование нужно проводить лишь для малой части банка, тем меньшей, чем больше длина слова W. Индексная таблица создаётся один раз для каждого банка и используется во всех поисках.

Недостаток — потеря чувствительности (можно пропустить достаточно сходные последовательности), тем бóльшая, чем больше W.

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 находок
  - 8 находок с E < 0.001
  - Время работы сервиса NCBI – менее одной минуты
- W = 2
  - Найдено 69 последовательностей, в них 75 находок
  - 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 ведется 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: $\mathbf{E\text{-}value = L_{query} \cdot L_{bank} \cdot K \cdot \exp(-\,Score \cdot \lambda)}$
  Score — это вес выравнивания, а параметры K и λ зависят от матрицы замен и штрафов за гэпы (их получили один раз для каждой матрицы и параметров штрафов путём вычислительного эксперимента)

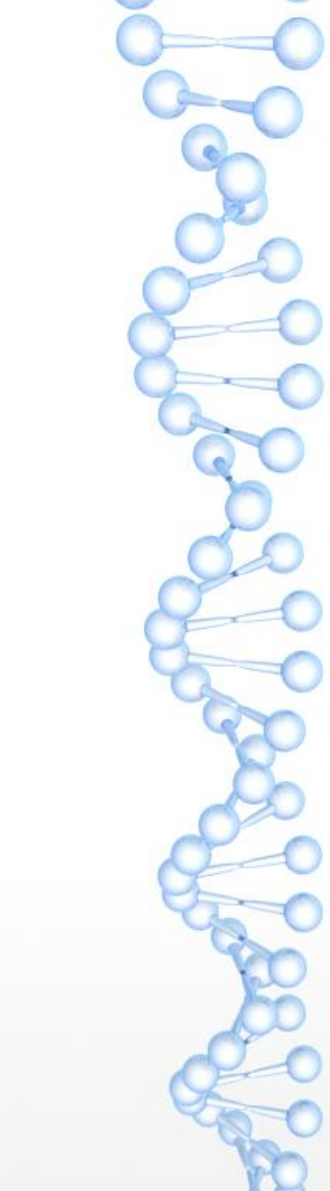
Нормализованный вес (вес в битах, bit score)

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

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

Интерпретация нормализованного веса не зависит от матрицы весов и штрафов за гэпы

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

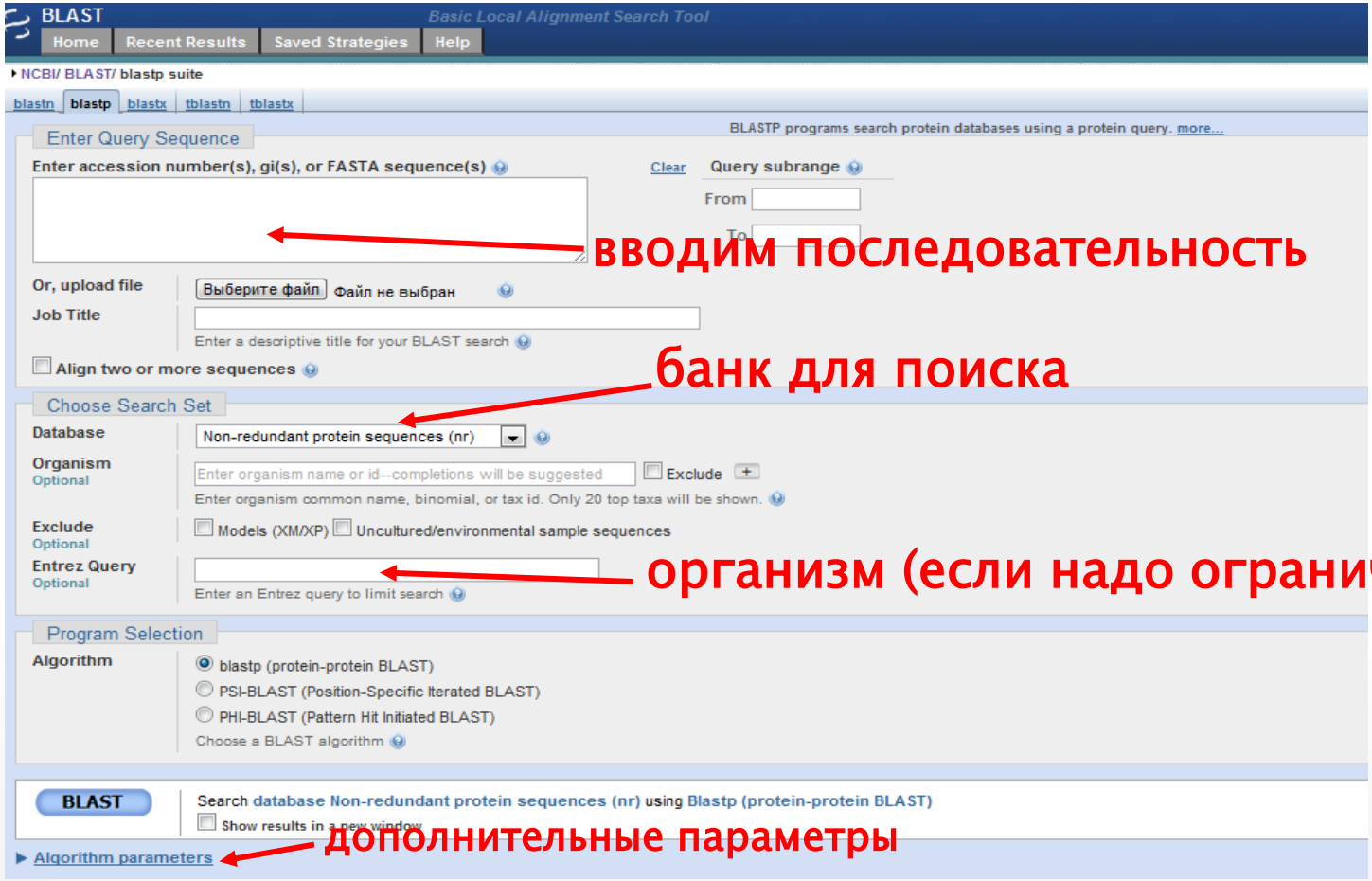


Задачи, которые решают с помощью BLAST

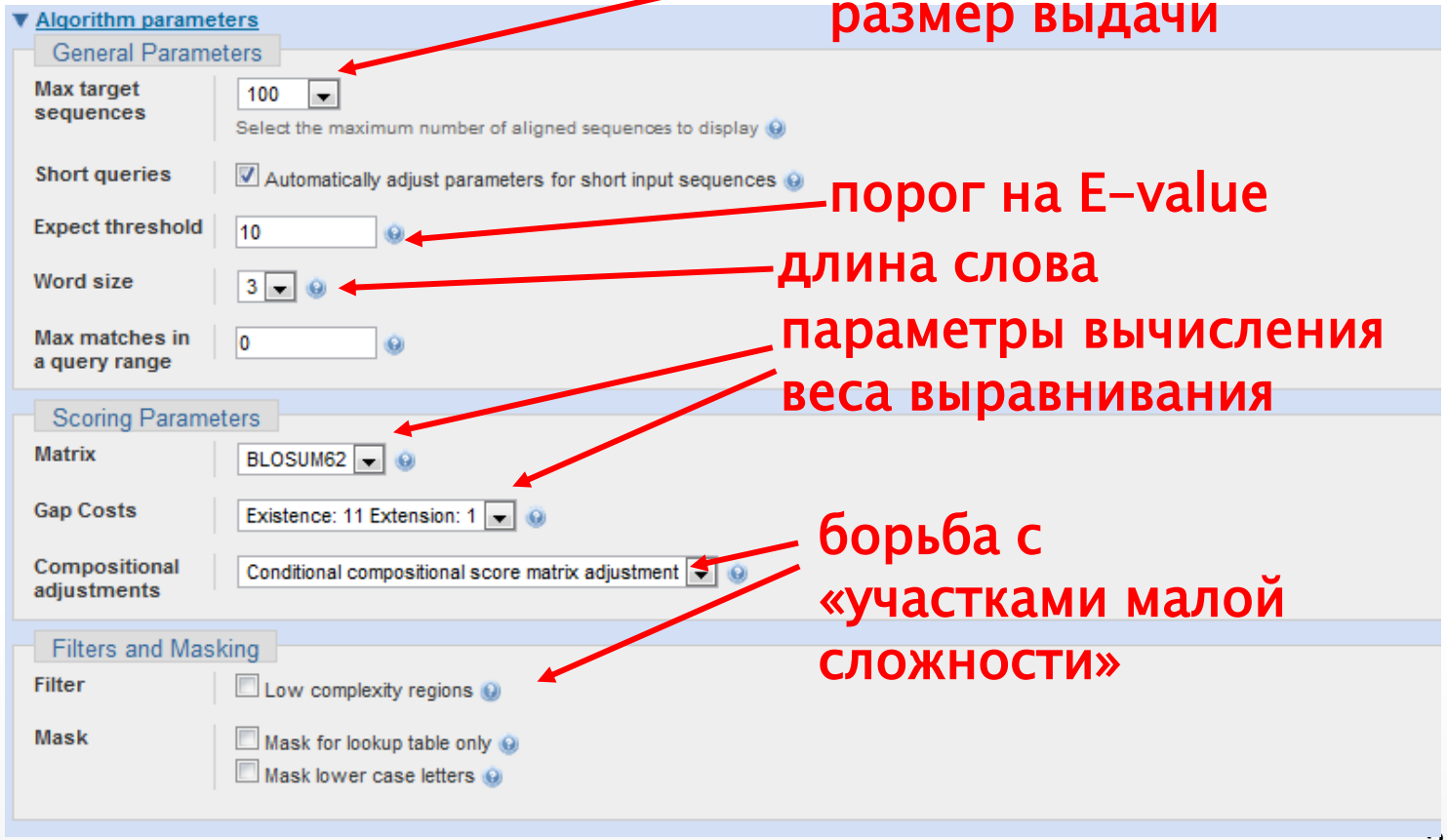
- Есть ли моя последовательность в банке?
- Есть ли у моего белка аннотированные гомологи? А гомологи с известной пространственной структурой?
- Хочу гомологи моего белка, чтобы:
  - предсказать его функцию *или*
  - понимать, какие его части консервативны $\Rightarrow$ важны для функции *или*
  - мало 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

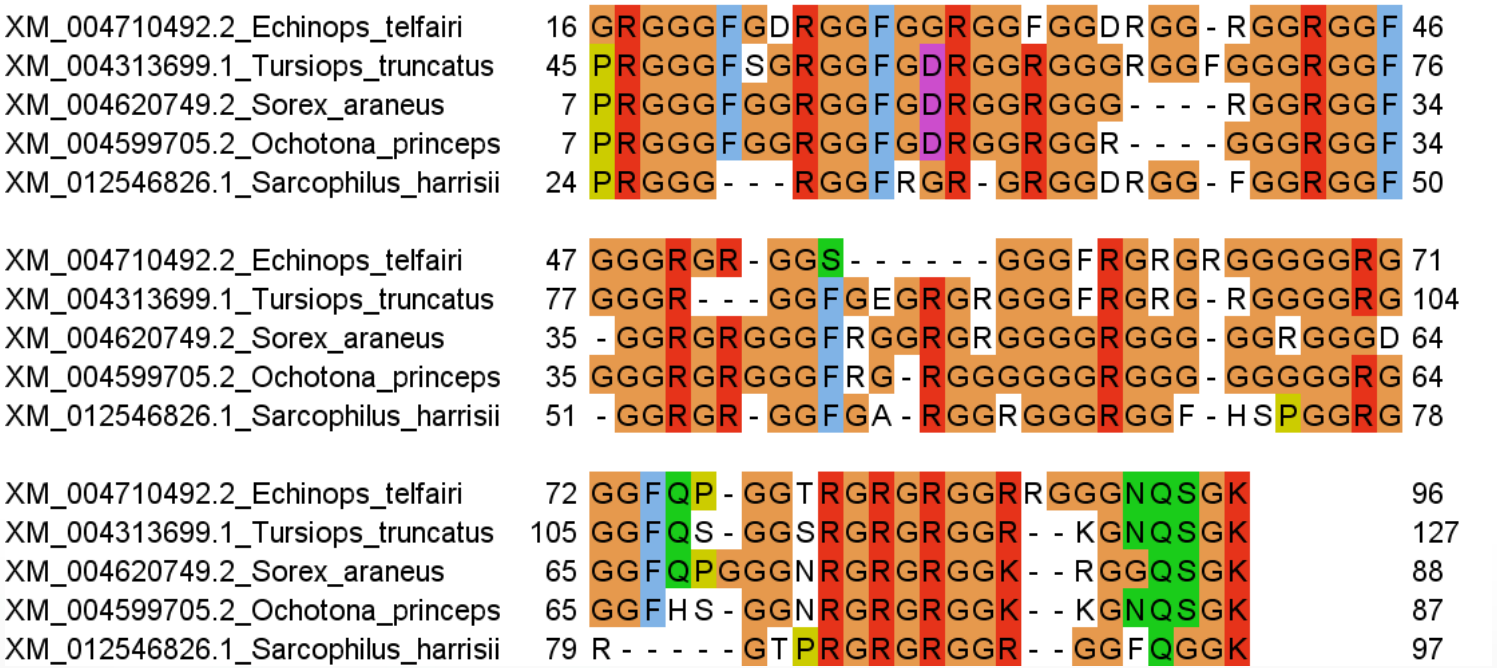
http://blast.ncbi.nlm.nih.gov/ → Protein blast



Интерфейс на сайте NCBI, дополнительные параметры



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



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,

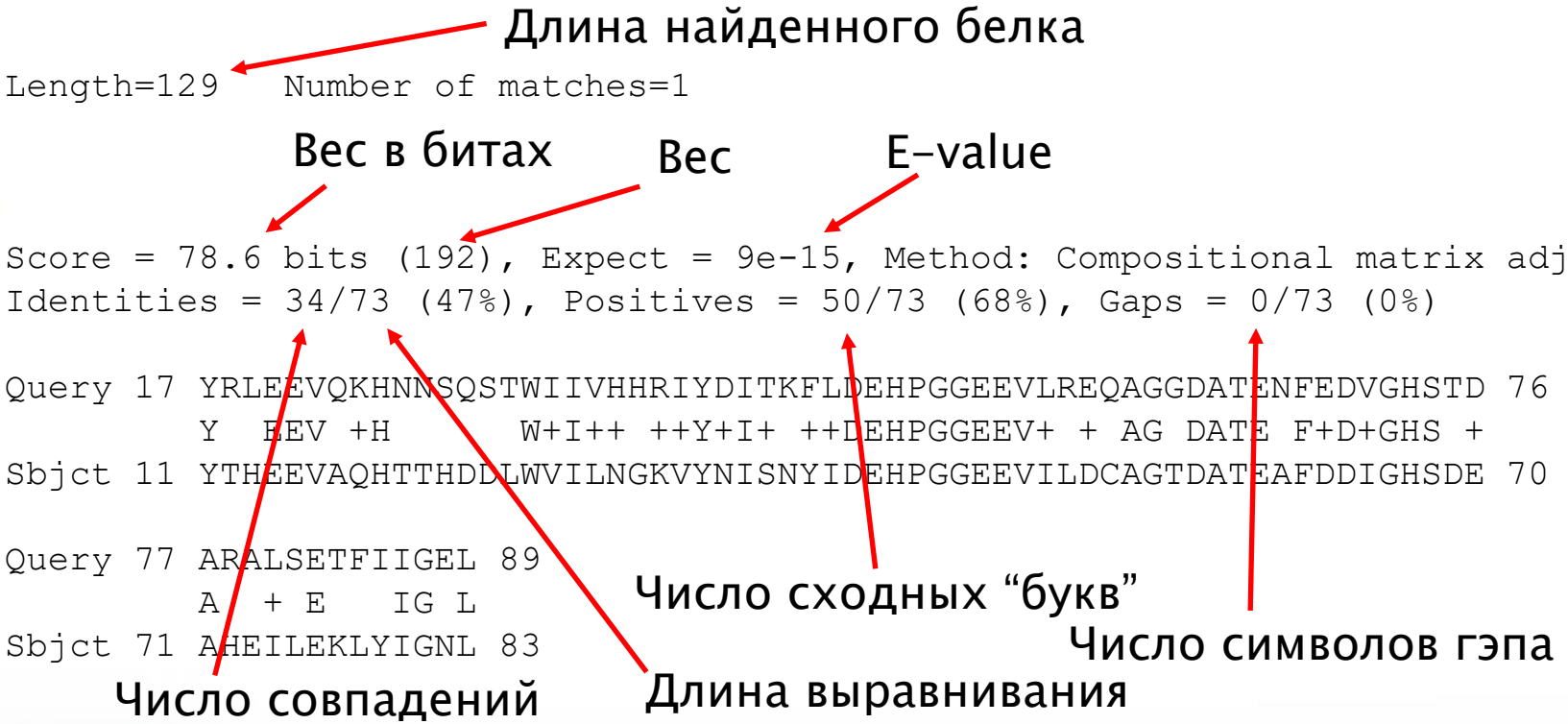
Выдача BLAST
(фрагмент — верхняя часть списка находок)

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	P49098.1
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



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 некодирующих 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 последовательнсоти, 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 6×6 = 36 сравнений для каждой банковской последовательности…

Больше подробностей — на сайте NCBI:
https://blast.ncbi.nlm.nih.gov/Blast.cgi?CMD=Web&PAGE_TYPE=BlastDocs