Практическая биоинформатика

Алгоритмы и программы множественного выравнивания Сравнение MSA

Алешина Юлия Александровна
Спирин Сергей Александрович

28 апреля 2026

Цель множественного выравнивания последовательностей гомологичных доменов или белков

- Реконструкция эволюции белков от общего предка. В колонке **правильного** выравнивания стоят аминокислотные остатки (а.о.), унаследованные от а.о. общего предка всех этих белков.
  Наследуются кодоны, это подразумевается.
- Основа для установления гомологии по сходству последовательностей.
  Сходство неравномерно на разных участках.

В какой мере выравнивание правильно — практически не проверяемо
(не считая парочки долговременных экспериментов по эволюции в лаборатории: Ленски — *E.coli*, Кондрашов — *Schizophyllum commune*, ...)

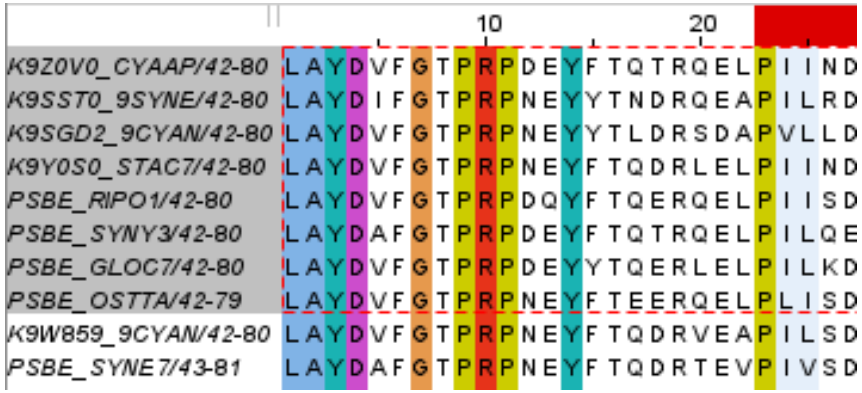
На практике стремятся к этой цели, но следует понимать, что она недостижима.

Разные программы множественного выравнивания используют разные алгоритмы и разные эвристики.

Как проверить правильность множественного выравнивания белков?

Искать блоки множественного выравнивания, построенного программой, которые соответствуют эволюционному?

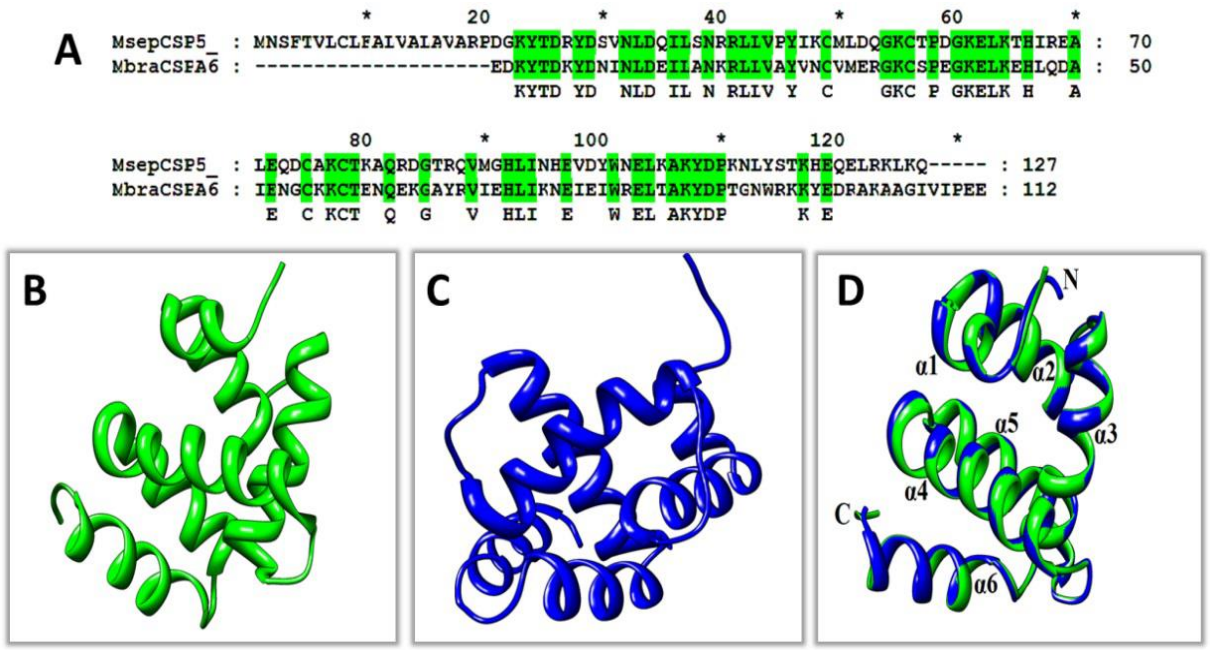
Крайне маловероятно, чтобы у нескольких белков было столько одинаковых букв на одинаковых расстояниях друг от друга по случайным причинам. Значит, скорее всего вы этом месте выравнивания правильное.



А если белки накопили очень много различий?

А если белки накопили очень много различий?

- 3D-структура белков консервативнее их последовательности!
- Совмещение структур
  - ✓ Cα-атомы а.о. из одной колонки при совмещении 3D структур этих белков оказываются в одном положении



Совмещение структур белков

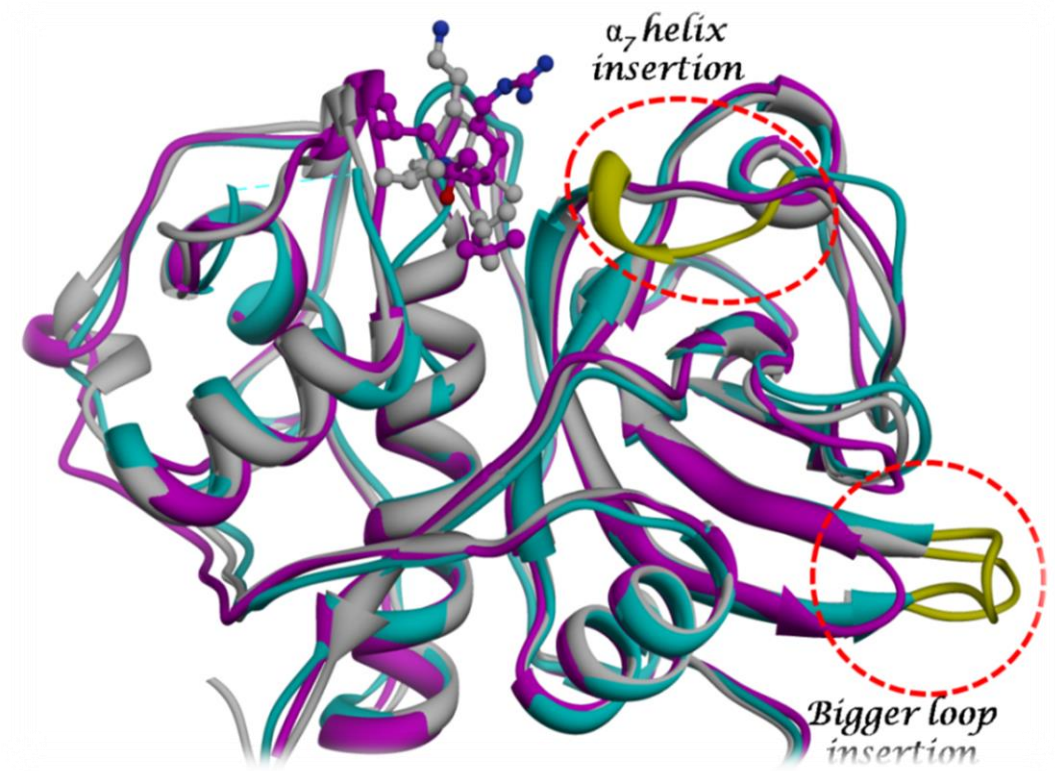
- Соответствие выравнивания совмещению 3D-структур можно проверить:
  - ✓ Накопить денег и сделать рентгеноструктурный анализ всех белков выравнивания
  - ✓ **Проверить на белках с известной структурой**
  - ✓ Предсказать 3D структуры всех белков и сравнить

  Основой предсказания служит сходство фрагментов последовательности с фрагментами белков с известной 3D-структурой (*даже при использовании AlphaFold, в конечном счёте*)

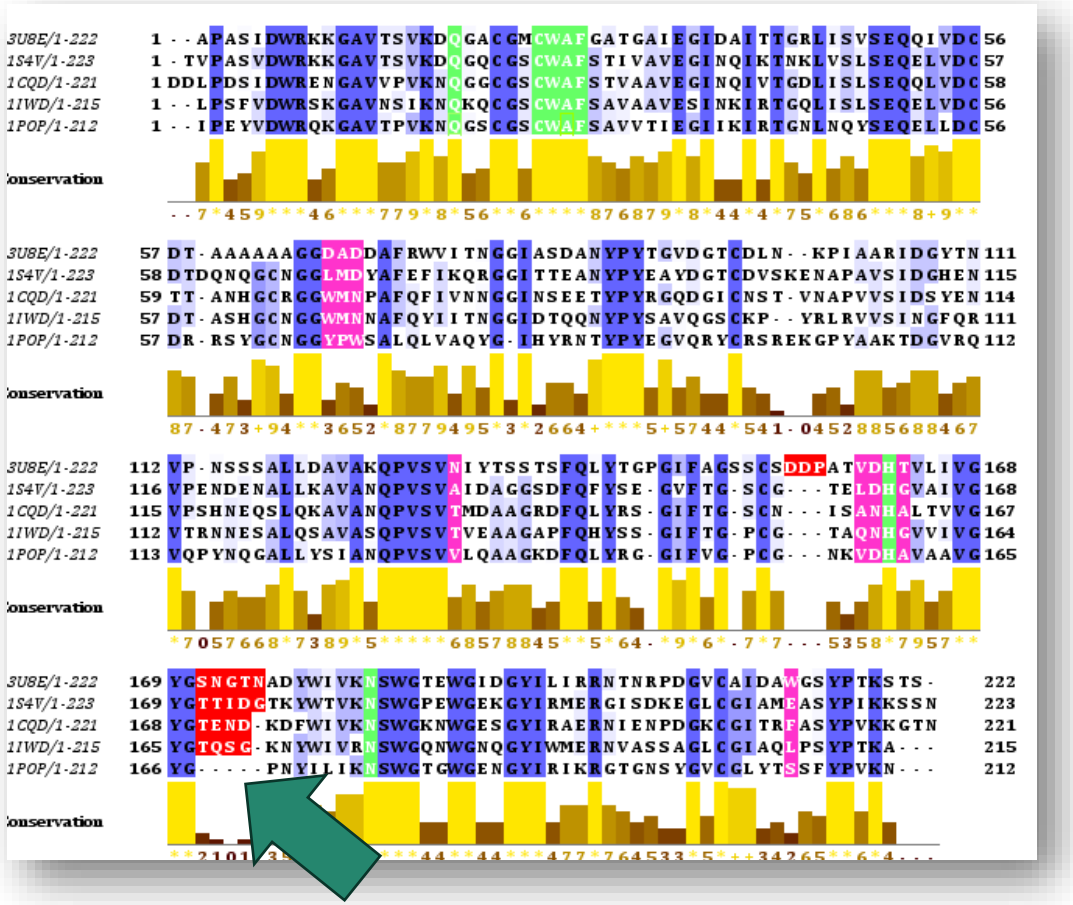
- Верно ли, что эволюция = сохранение хода полипептидной цепи в 3D?

Эволюция = сохранение хода полипептидной цепи в 3D?

- Отчасти, да – есть много примеров совмещаемых структур со слабым сходством последовательностей
- Но сравнительно длинные вставки в одном и том же месте структуры могут произойти независимо и не быть гомологичными



Выравнивание по совмещению структур папаин-подобных цистеиновых протеаз растений



-- **Conserved**

-- **Catalytic**

-- **Active site**

-- *3U8E* (*Crocus sativus*)
-- 1S4V (*Ricinus communis*)
-- 1CQD (*Zingiber officinale*)
-- 1IWD (Ervatamin B)
-- 1POP (*Carica papaya*)

В этих колонках мы не уверены, что амк. остатки гомологичны

Совмещение структур белков

- PDBeFold – и парное, и множественное совмещение

https://www.ebi.ac.uk/msd-srv/ssm/cgi-bin/ssmserver

- TM-Align, FATCAT и др. – парное сравнение, веб-сервис доступен на сайте RSCB PDB

https://www.rcsb.org/alignment

- Caretta – множественное совмещение

https://github.com/TurtleTools/caretta

Алгоритмы и программы множественного выравнивания

Динамическое программирование

Алгоритм Нидлмана — Вунша для **парного** выравнивания

Время работы и требуемая память пропорциональны произведению длин последовательностей (пишут “сложность O(nm)”)

match = 1 mismatch = -1 gap = -1

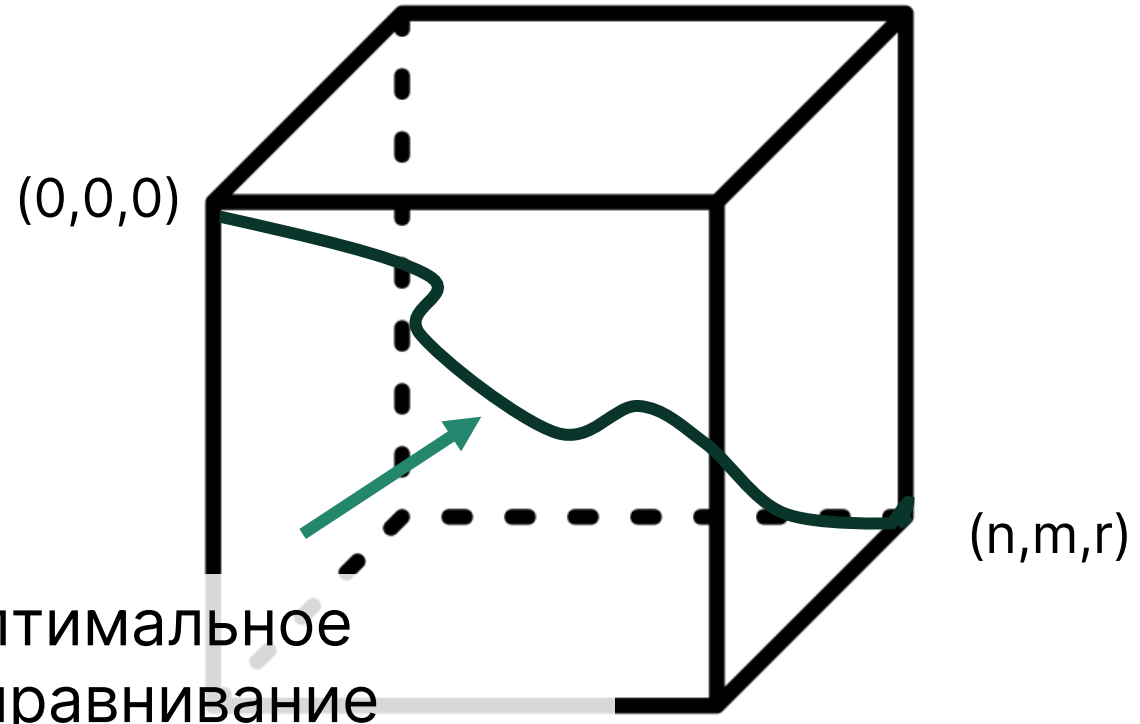
		G	C	A	T	G	C	G
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	1	0	-1	-2	-3	-4	-5
A	-2	0	0	1	0	-1	-2	-3
T	-3	-1	-1	0	2	1	0	-1
T	-4	-2	-2	-1	1	1	0	-1
A	-5	-3	-3	-1	0	0	0	-1
C	-6	-4	-2	-2	-1	-1	1	0
A	-7	-5	-3	-1	-2	-2	0	0

Динамическое программирование

match = 1 mismatch = -1 gap = -1

		G	C	A	T	G	C	G
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	1	0	-1	-2	-3	-4	-5
A	-2	0	0	1	0	-1	-2	-3
T	-3	-1	-1	0	2	1	0	-1
T	-4	-2	-2	-1	1	1	0	-1
A	-5	-3	-3	-1	0	0	0	-1
C	-6	-4	-2	-2	-1	-1	1	0
A	-7	-5	-3	-1	-2	-2	0	0

Множественное для трех последовательностей

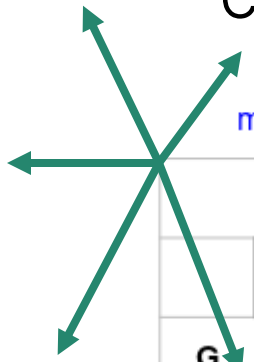


Динамическое программирование

match = 1 mismatch = -1 gap = -1

		G	C	A	T	G	C	G
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	1	0	-1	-2	-3	-4	-5
A	-2	0	0	1	0	-1	-2	-3
T	-3	-1	-1	0	2	1	0	-1
T	-4	-2	-2	-1	1	1	0	-1
A	-5	-3	-3	-1	0	0	0	-1
C	-6	-4	-2	-2	-1	-1	1	0
A	-7	-5	-3	-1	-2	-2	0	0

Множественное выравнивание
Сложность $O(n_1 * n_2 * \ldots * n_k)$



match = 1 mismatch = -1 gap = -1

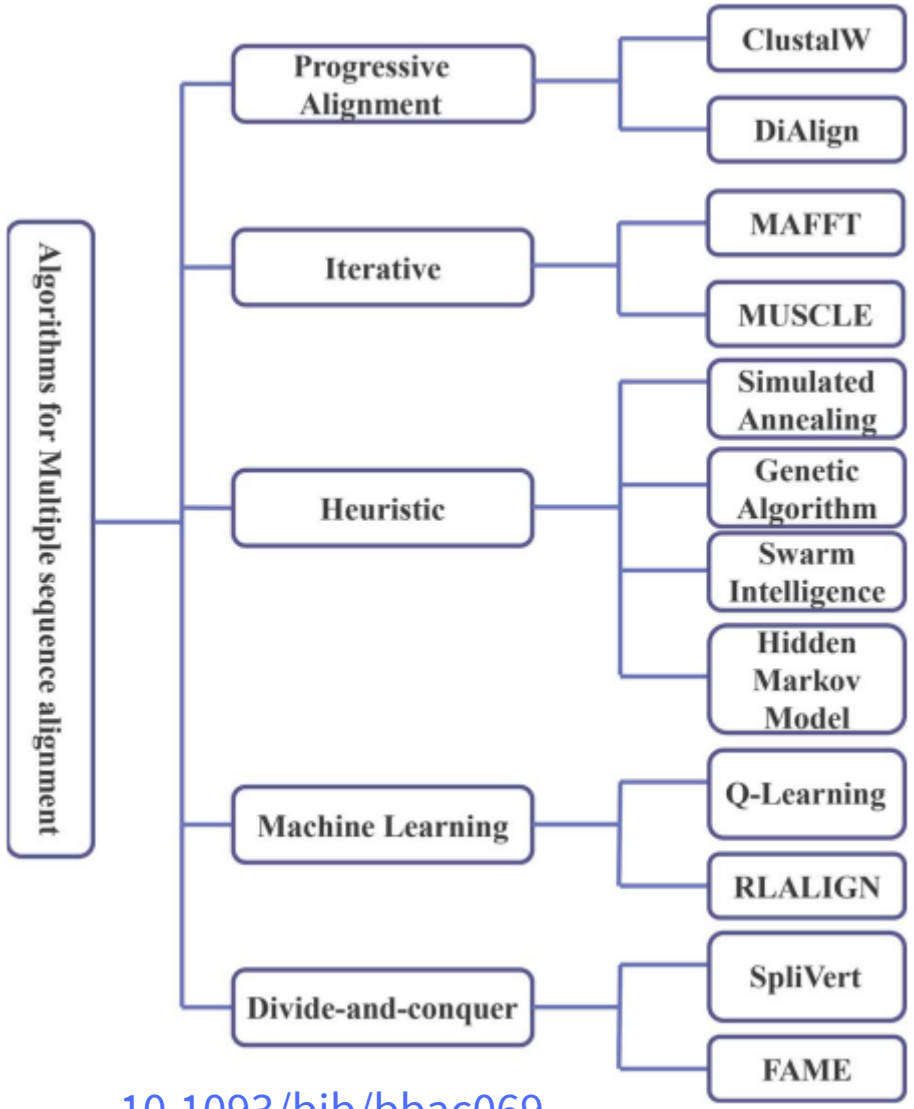
		G	C	A	T	G	C	G
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	1	0	-1	-2	-3	-4	-5
A	-2	0	0	1	0	-1	-2	-3
T	-3	-1	-1	0	2	1	0	-1
T	-4	-2	-2	-1	1	1	0	-1
A	-5	-3	-3	-1	0	0	0	-1
C	-6	-4	-2	-2	-1	-1	1	0
A	-7	-5	-3	-1	-2	-2	0	0

Нахождение оптимального выравнивания для большого числа последовательностей практически неосуществимо

Выход — эвристические алгоритмы

Программы множественного выравнивания

- Программ много, и все они используют разные **эвристические** алгоритмы
- В литературе встречаются разные их классификации (например, https://link.springer.com/protocol/10.1007/978-1-0716-1036-7_17)



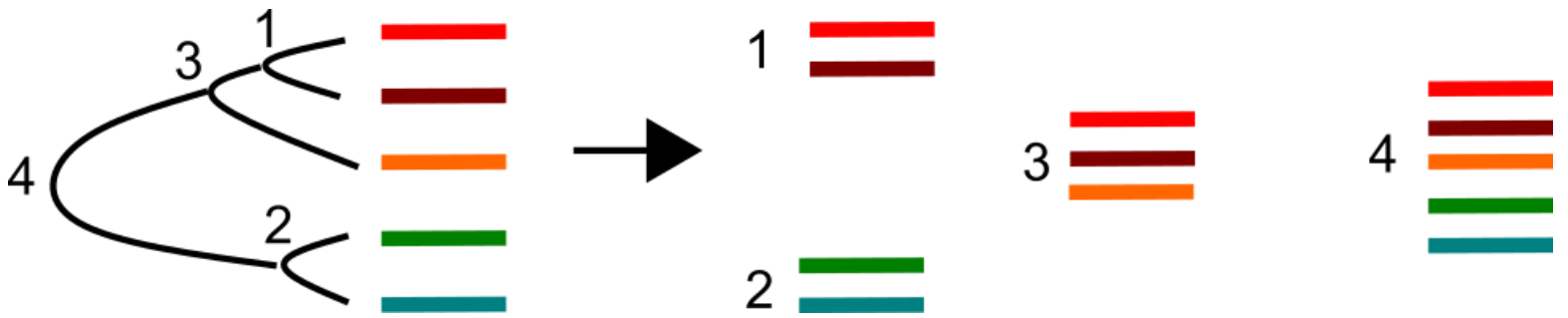
10.1093/bib/bbac069

Список популярных программ MSA

Program	Год	В Jalview	На kodomo	Website
ClustalO	2011	Yes	No	https://www.ebi.ac.uk/jdispatcher/msa/clustalo
ClustalW	1994	Yes	Yes	http://www.clustal.org/
MAFFT	2002	Yes	Yes	https://mafft.cbrc.jp/alignment/software/
MSAProbs	2010	Yes	No	http://msaprobs.sourceforge.net/
MUSCLE	2004	Yes	Yes	http://www.drive5.com/muscle/
ProbCons	2005	Yes	No	http://probcons.stanford.edu/
T-Coffee	2000	Yes	No	https://tcoffee.crg.eu/
Glprobs	2015	Yes	No	https://sourceforge.net/projects/glprobs/
DiAlign	1998	No	Yes	http://dialign.gobics.de/
FAME	2020	No	No	http://github.com/naznoosh/msa
Kalign	2005	No	No	https://www.ebi.ac.uk/Tools/msa/kalign/
NX4	2019	No	No	https://www.nx.io
PRANK	2008	No	Yes	http://wasabiapp.org/software/prank/
Probalign	2006	No	No	http://probalign.njit.edu/standalone.html

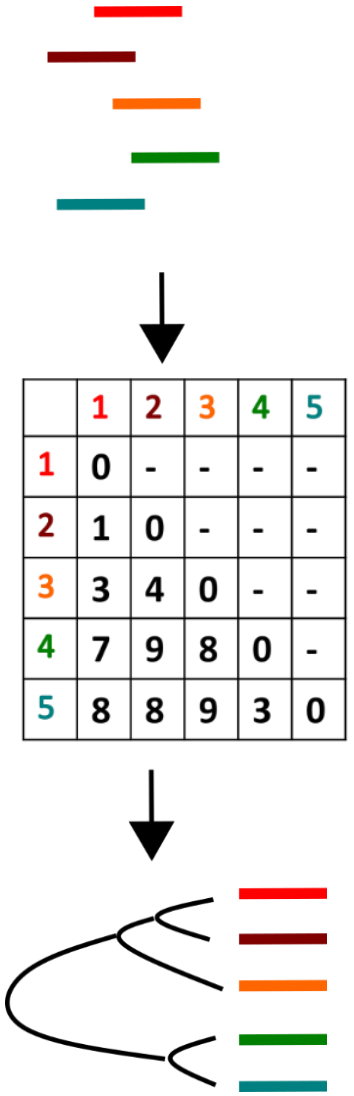
Прогрессивное выравнивание

- Основная идея: *выравнивание двух выравниваний* с помощью динамического программирования
- Этапы:
  - ✓ Построить дерево родственности всех последовательностей – направляющее дерево
  - ✓ Выравниваем стопки выровненных последовательностей от листьев до корня



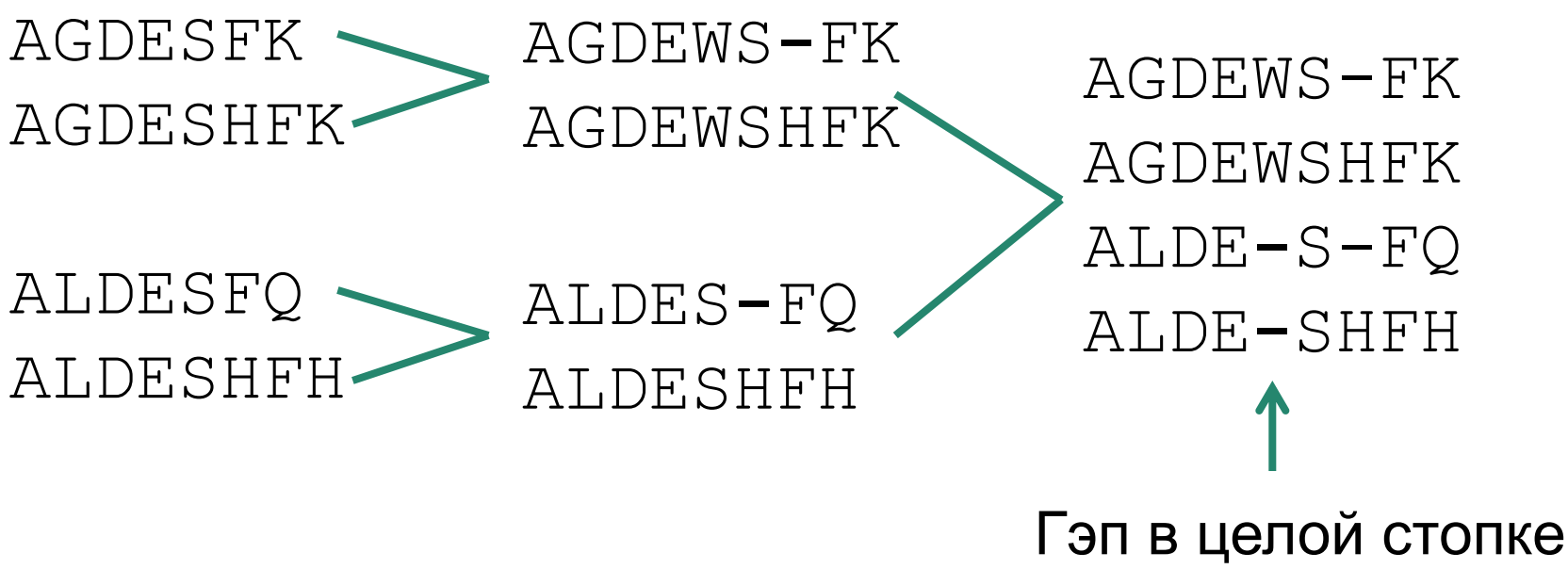
Построение направляющего дерева

- Для ВСЕХ ПАР последовательностей строится парное выравнивание.
- Вес парного выравнивания пересчитывается в расстояние между последовательностями:
  - ✓ чем больше вес, тем меньше расстояние;
  - ✓ расстояние между совпадающими последовательностями равно 0.
- Получается матрица расстояний между послед-ми
- Есть алгоритмы, превращающие матрицу попарных расстояний в дерево.
  - ✓ Расстояния между листьями по дереву отражают сходство последовательностей

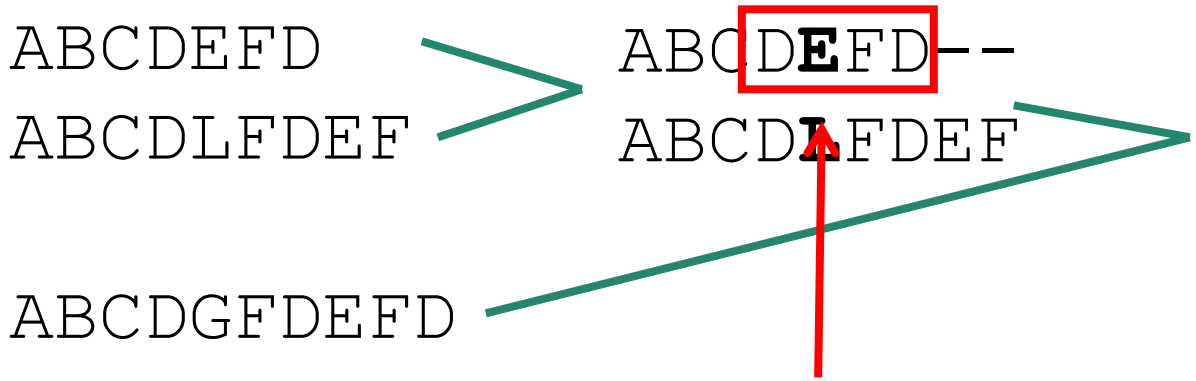


	1	2	3	4	5
1	0	-	-	-	-
2	1	0	-	-	-
3	3	4	0	-	-
4	7	9	8	0	-
5	8	8	9	3	0

Пример прогрессивного выравнивания



Проблема: если ошиблись, то ошибка никуда не денется



Но уже поздно передвигать этот блок

Получилось

```
ABCDEFD---
ABCDLFDEF-
ABCDGFDEFD
```

А хотелось бы так:

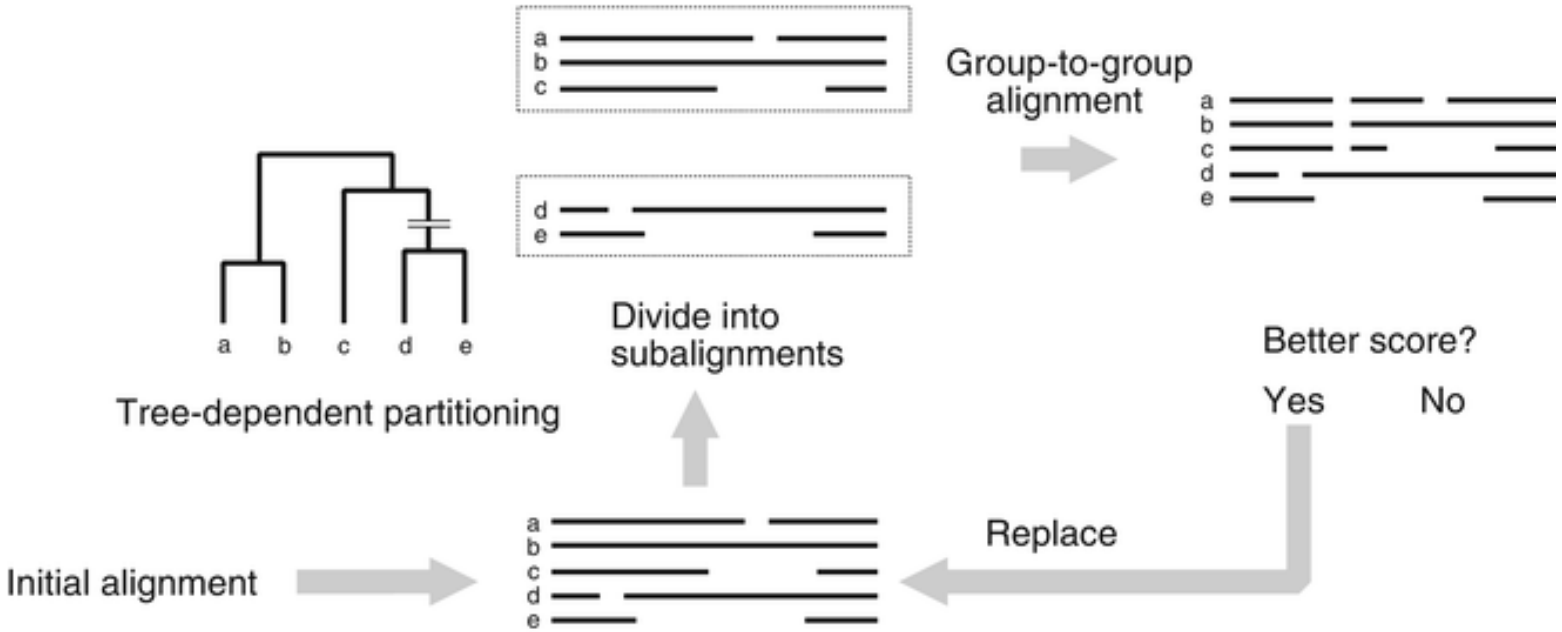
```
ABC---DEFD
ABCDLFDEF-
ABCDGFDEFD
```

Итеративное улучшение (refinement) выравнивания

i. Построить множественное выравнивание

ii. Разделить его на две группы

iii. Перевыровнять две группы. Повторить ii



Иерархический алгоритм выравнивания многих последовательностей

- Основная идея: *выравнивание двух выравниваний* с помощью динамического программирования
- Этапы алгоритма
  - ✓ Построение направляющего дерева
  - ✓ Итерация выравнивания выравниваний
  - ✓ “Рафинирование” (refinement) выравнивания
- Результат – ГЛОБАЛЬНОЕ множественное выравнивание

Список популярных программ множественного выравнивания

- Прогрессивное выравнивание (ClustalW)
- Прогрессивное выравнивание →итеративное улучшение (Muscle, PRALINE, IterAlign, MAFFT)
- Методы, учитывающие согласованность попарных сравнений (PROBCONS, Prank, Muscle5)
- Методы, использующие структурные данные (3D-Coffee)

А какая программа лучше?

- Все алгоритмы множественного выравнивания эвристические

Как сравнить программы?

- Базы данных идеальных выравниваний!
  Обычно 3D-наложение структур + ручная проверка экспертами
- BAliBASE (Benchmark Alignment dataBASE) – наиболее признанная
  - ✓ 10 наборов референсных выравниваний, покрывающих разные особенности (высокая вариабельность, повторы и т д)
- Есть и другие
  - ✓ SABMark(SequenceAlignmentBenchMark)
  - ✓ OXBench(OXfordBenchmark)
  - ✓ SMART(SimpleModularArchitectureResearchTool)
  - ✓ PREFAB(ProteinREFerenceAlignmentBenchmark)
  - ✓ и др.

Результаты сравнения программ на BaliBase

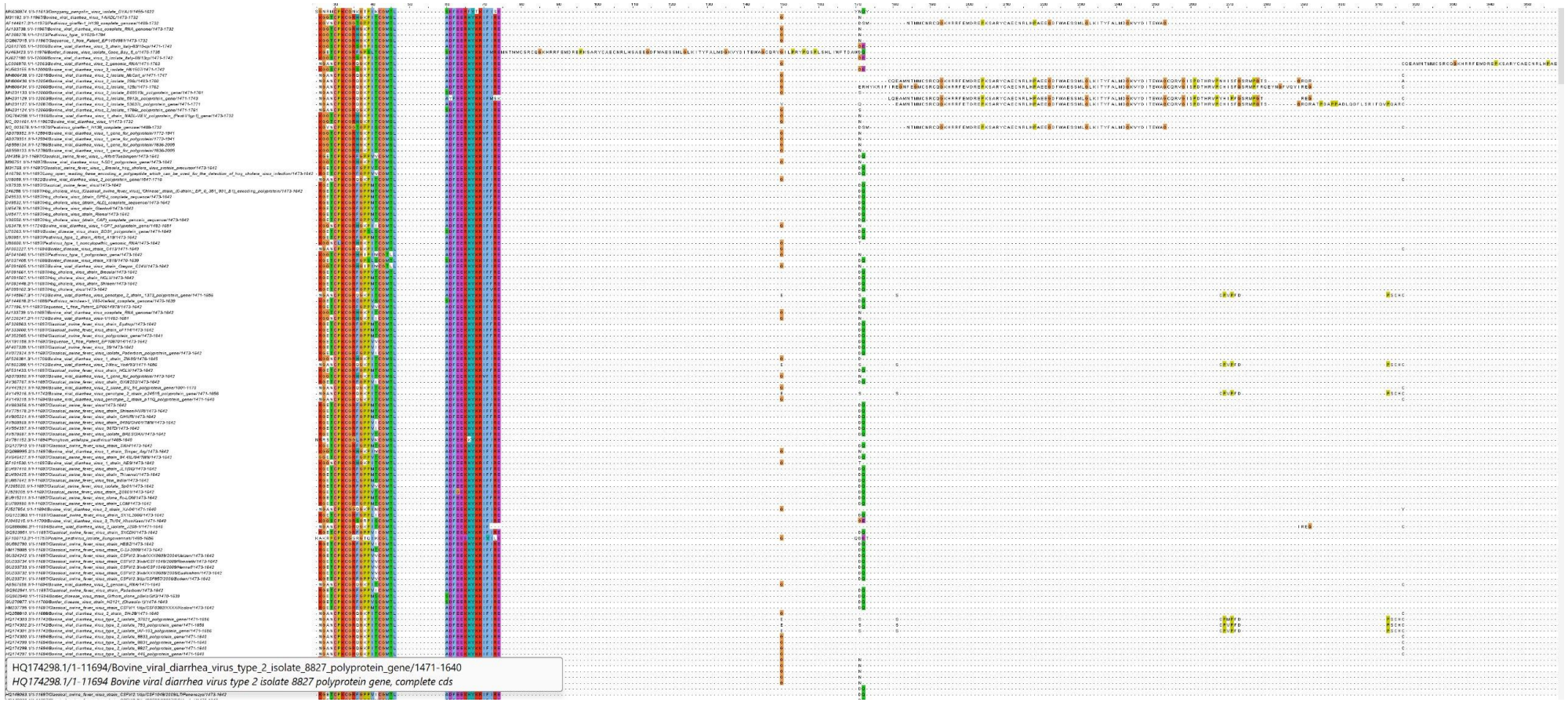
Program	R1-1	R1-2	R2	R3	R4	R5	Avg Score	Tot time(s)
MSAProbs [122]	0.441	**0.865**	**0.464**	**0.607**	**0.622**	**0.608**	**0.607**	12382.00
Probalign [84]	**0.453**	0.862	0.439	0.566	0.603	0.549	0.589	10095.20
MAFFT [14]	0.439	0.831	0.450	0.581	0.605	0.591	0.588	1475.40
Probcons [17]	0.417	0.855	0.406	0.544	0.532	0.573	0.558	13086.30
Clustal Omega [123]	0.358	0.789	0.450	0.575	0.579	0.533	0.554	539.91
T-Coffee [71]	0.410	0.848	0.402	0.491	0.545	0.587	0.551	81041.50
Kalign [72]	0.365	0.790	0.360	0.476	0.504	0.435	0.501	**21.88**
MUSCLE [15]	0.318	0.804	0.350	0.409	0.450	0.460	0.475	789.57
FSA [136]	0.270	0.818	0.187	0.259	0.474	0.398	0.419	53648.10
DiAlign [12]	0.265	0.696	0.292	0.312	0.441	0.425	0.415	3977.44
PRANK [137]	0.223	0.680	0.257	0.321	0.360	0.356	0.376	128355.00
ClustalW [11]	0.227	0.712	0.220	0.272	0.396	0.308	0.374	766.47

- Колонки 2-7 – разные наборы тестовых выравниваний.
- Указан TC (total column) вес сравнения с референсным выравниванием.
- Значения от 0 до 1. Чем больше – тем более похоже выравнивание на референсное.
- Последняя колонка – время работы программы

10.1093/bib/bbac069

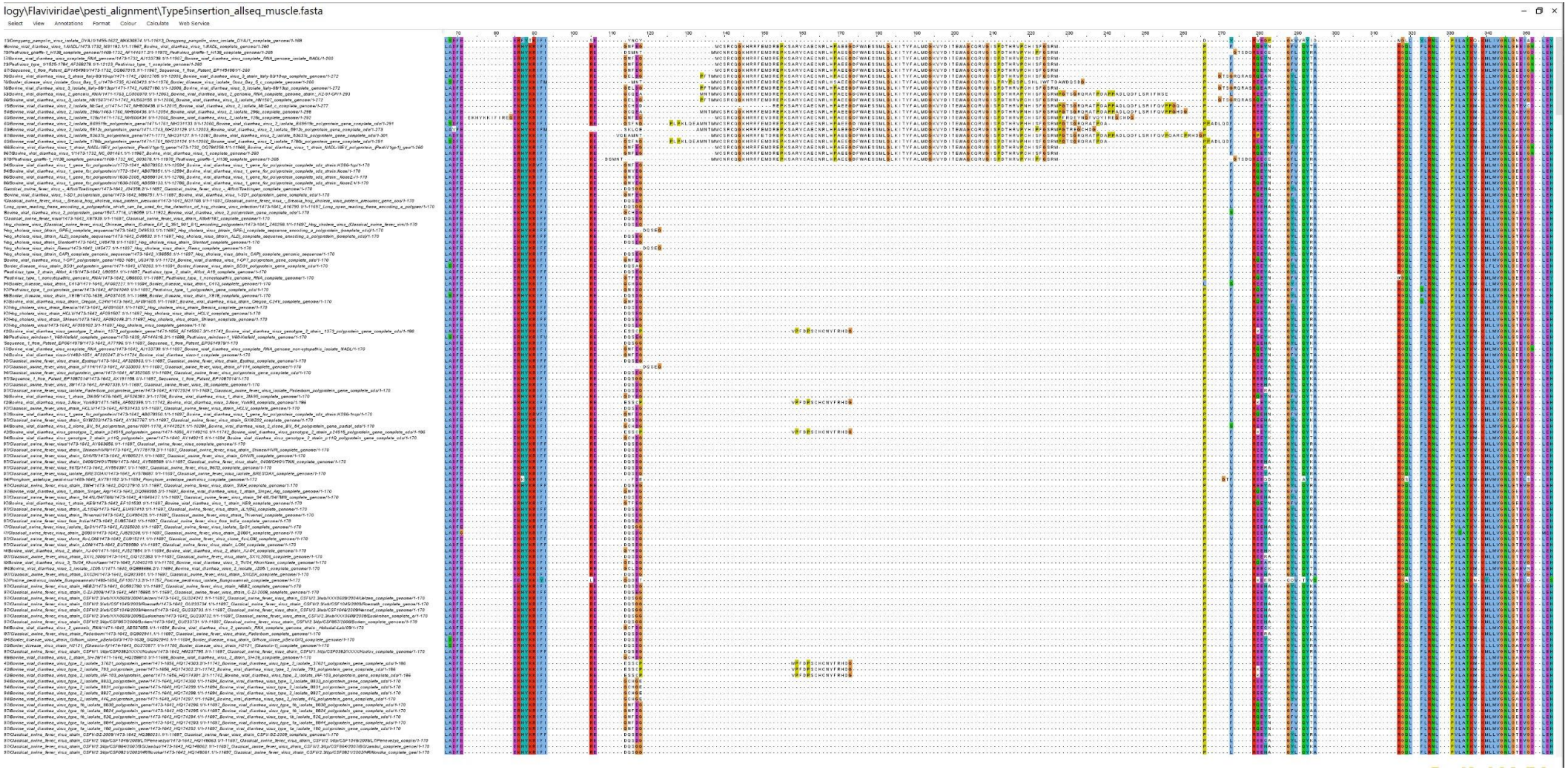
Результаты выравнивания одних и тех же последовательностей разными программами могут существенно отличаться!

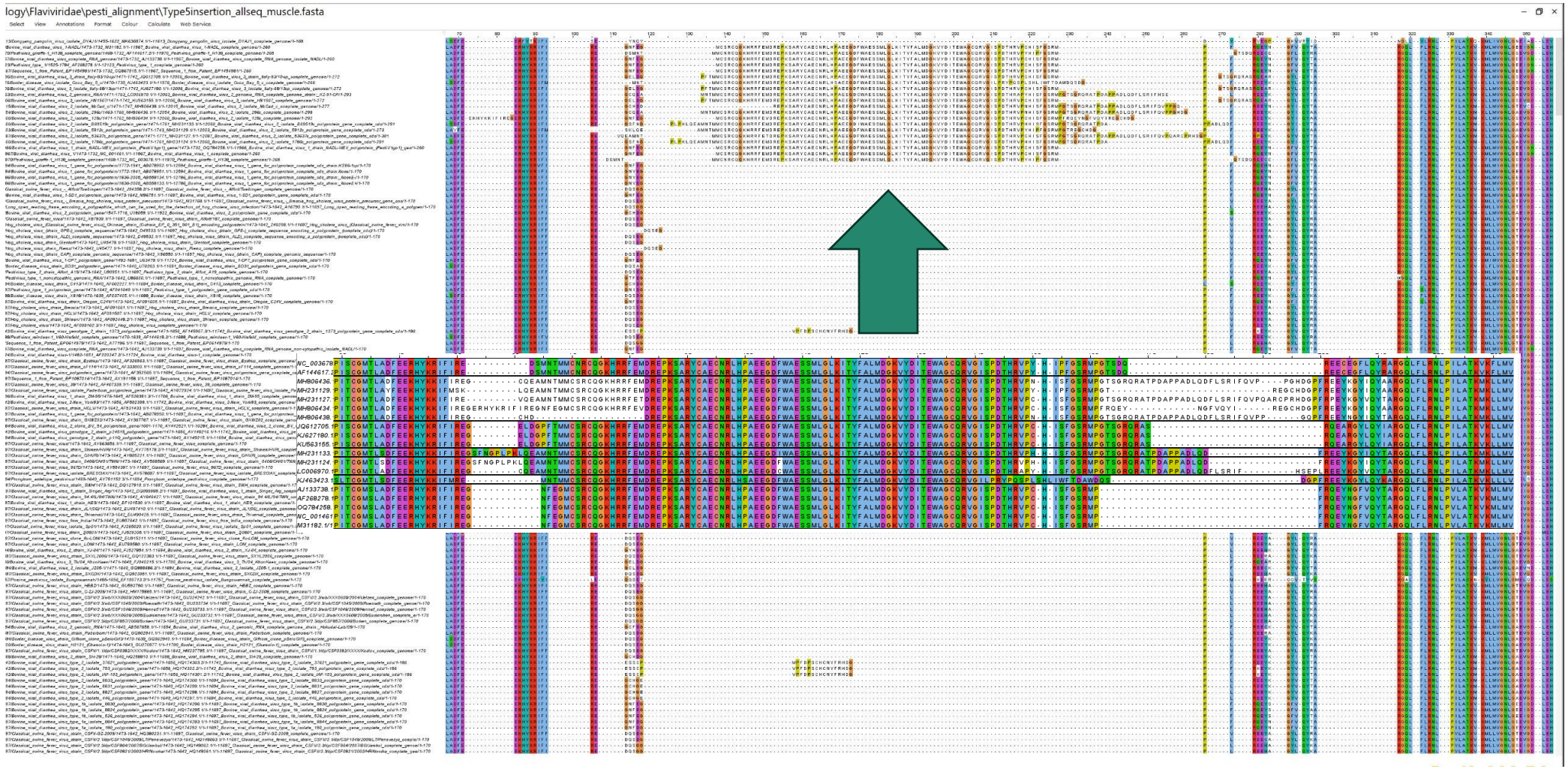
Пример из жизни. Выравнивание участка с вставкой Jiv90 полипротеинов пестивирусов.



MAFFT

Выравнивание тех же последовательностей с помощью MUSCLE




logy\Flaviviridae\pesti_alignment\Type5insertion_allseq_muscle.fasta
Select View Annotations Format Colour Calculate Web Service

Сравнение разных выравниваний одних и тех же последовательностей

Построенных разными алгоритмами множественного выравнивания

Какие выравнивания тех же последовательностей совпадают?

1

```
Seq1 MKFR-SSHYA-S
Seq2 MKYRRS-HYA-S
Seq3 MEFRRRSHYA-R
```

2

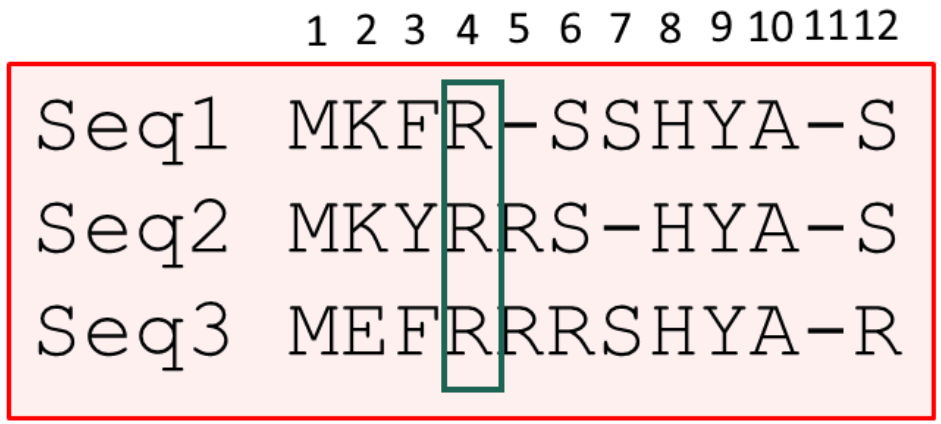
```
Seq1 MKF-RSSHYAS
Seq2 MKYRRS-HYAS
Seq3 MEFRRRSHYAR
```

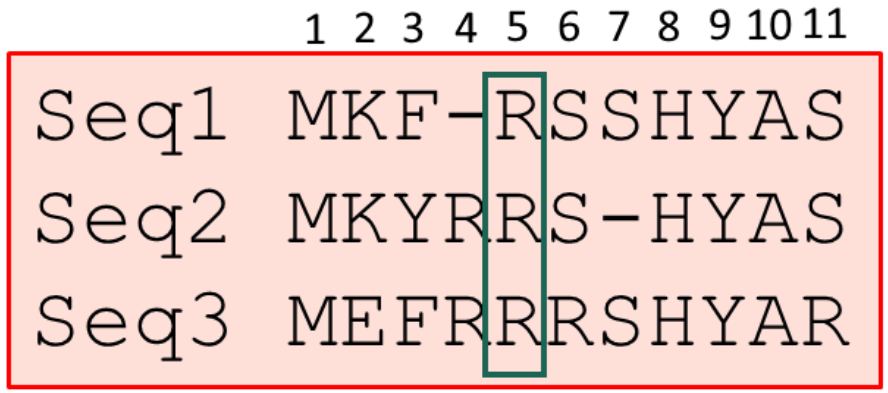
4

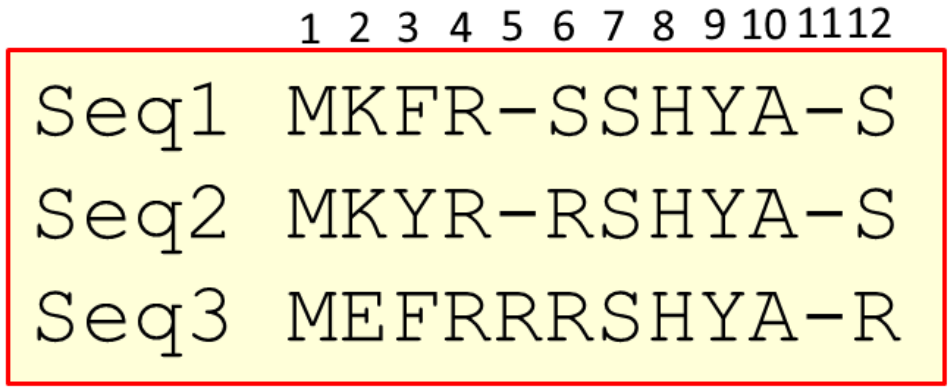
```
Seq1 MKFR-SSHYA-S
Seq2 MKYR-RSHYA-S
Seq3 MEFRRRSHYA-R
```

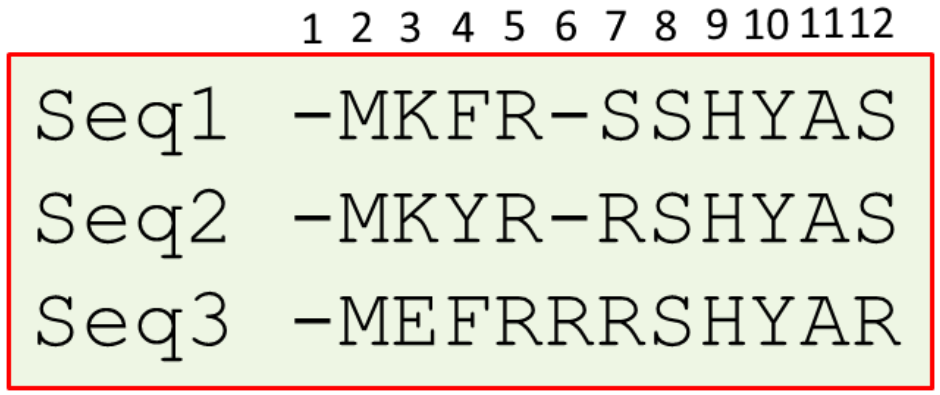
3

```
Seq1 -MKFR-SSHYAS
Seq2 -MKYR-RSHYAS
Seq3 -MEFRRRSHYAR
```









Колонка i выравнивания X совпадает с колонкой j выравнивания Y если в них – те же самые остатки;
те же самые значит – с теми же номерами, а не с теми же буквами!

```
Seq1 MKFR-SSHYA-S
Seq2 MKYRRS-HYA-S
Seq3 MEFRRRSHYA-R
```

```
Seq1 MKF-RSSHYAS
Seq2 MKYRRS-HYAS
Seq3 MEFRRRSHYAR
```

Вместо аминокислоты укажем номер остатка в последовательности:

```
Seq1 1234-5678 9 -10
Seq2 123456-78 9 -10
Seq3 123456789 10-11
```

```
Seq1 123-45678 9  10
Seq2 123456-78 9  10
Seq3 123456789 10 11
```

```
Seq1 MKFR-SSHYA-S
Seq2 MKYRRS-HYA-S
Seq3 MEFRRRSHYA-R
```

```
Seq1 MKF-RSSHYAS
Seq2 MKYRRS-HYAS
Seq3 MEFRRRSHYAR
```

Вместо аминокислоты укажем номер остатка в последовательности:

```
Seq1 1234-5678 9 -10
Seq2 123456-78 9 -10
Seq3 123456789 10-11
```

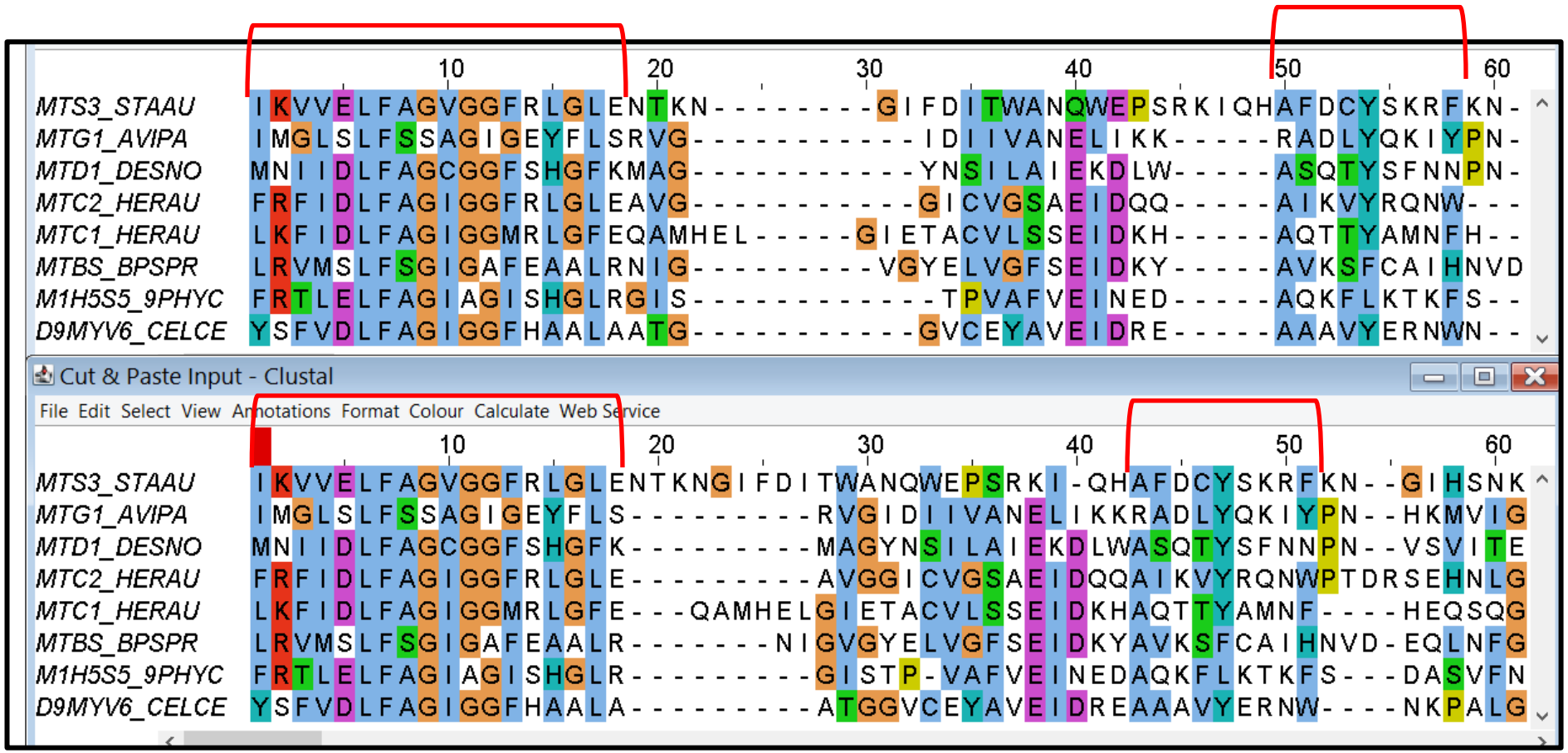
```
Seq1 123-45678 9  10
Seq2 123456-78 9  10
Seq3 123456789 10 11
```

Точное сравнение двух выравниваний

- Два выравнивания I и II тех же последовательностей совпадают если в каждой колонке i (i = 1, 2, …, N) выравниваний I и II стоят те же самые буквы.
- Буквы не в смысле а.к.о., а в смысле номера буквы в последовательности
- Различие выравниваний I и II определяется числом колонок, для которых это не так и их расположением

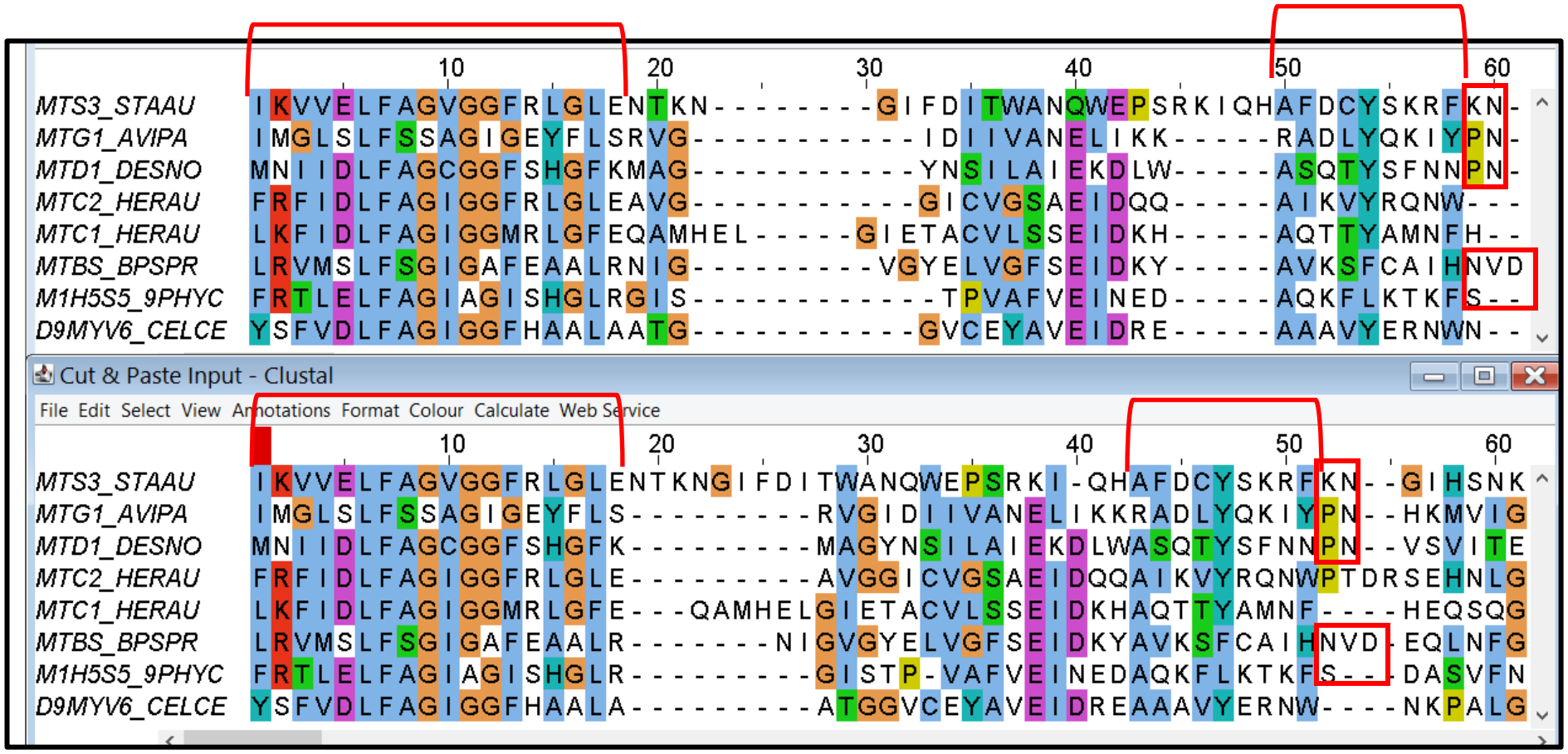
Выравнивания совпадают на двух выделенных участках

Ещё есть совпадения не на всех, а на подмножествах последовательностей. НАЙДИТЕ!

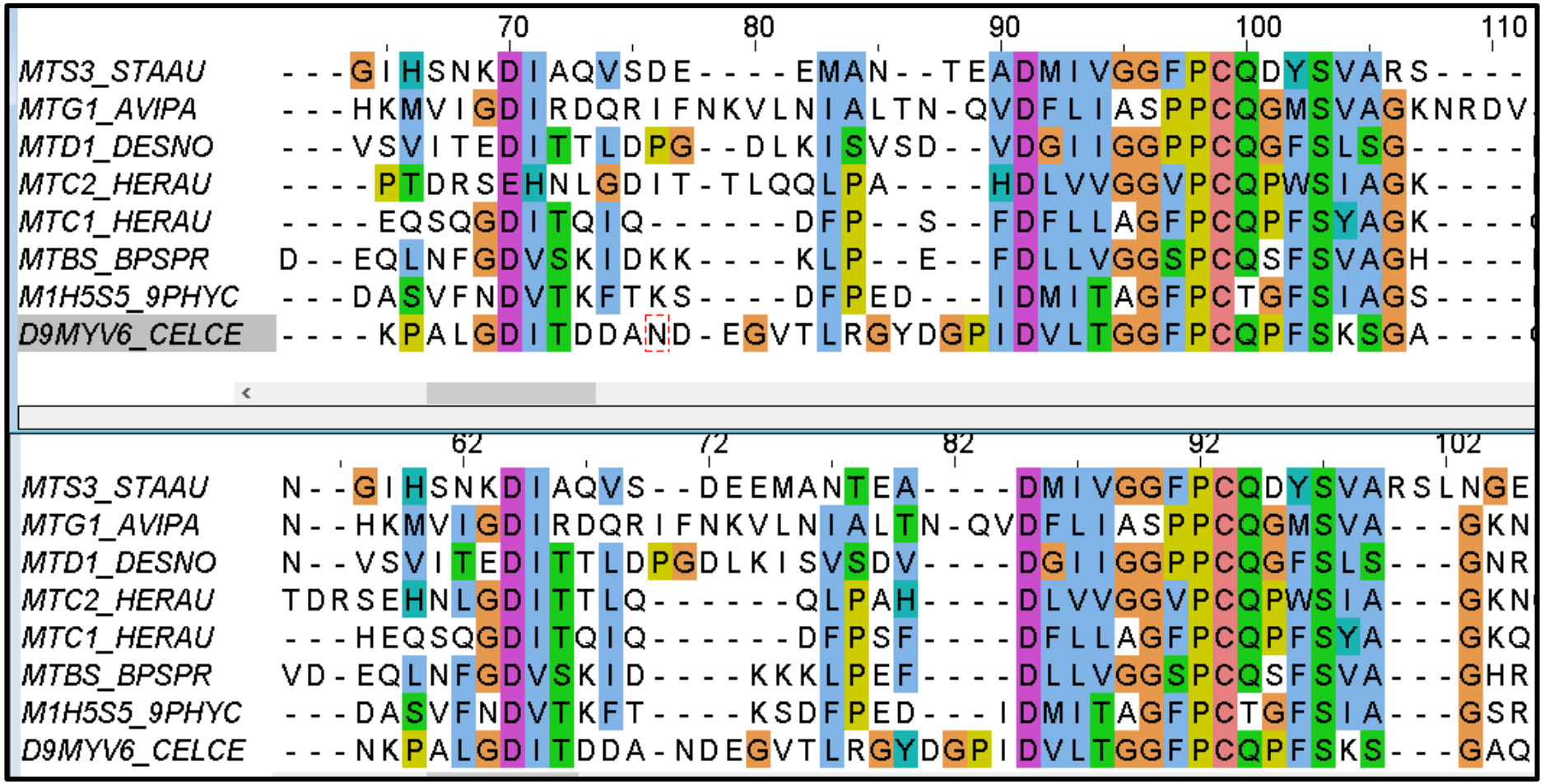


Выравнивания совпадают на двух выделенных участках

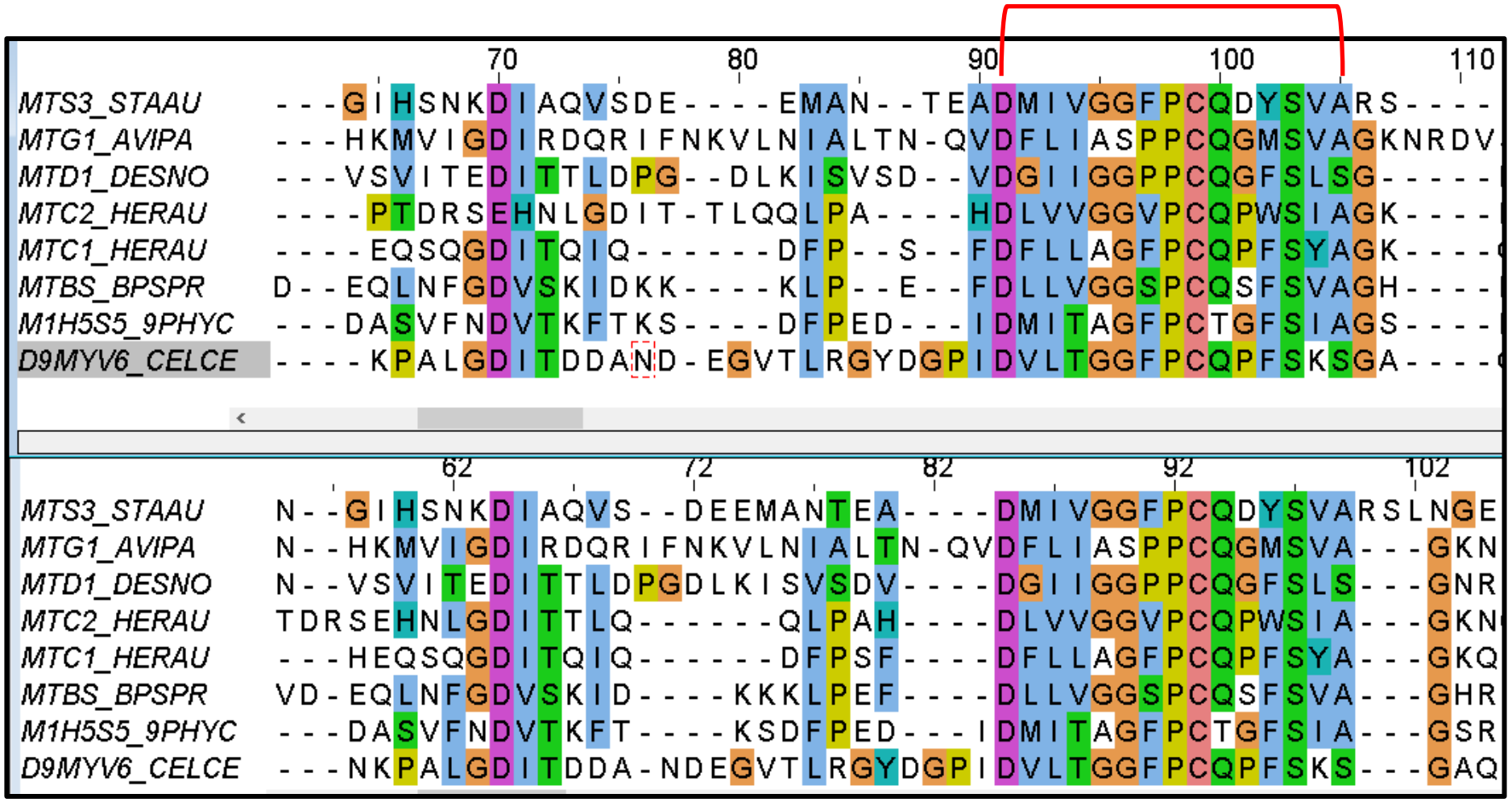
Ещё есть совпадения не на всех, а на подмножествах последовательностей. НАЙДИТЕ!



Продолжение. Разметьте самостоятельно!



Если в двух блоках без гэпов из разных выравниваний есть хотя бы одна одинаково выровненная позиция, то блоки выровнены одинаково



Как сравнить два множественных выравнивания

1. Отсортировать последовательности по ID в обоих выравниваниях.
2. Если колонка (номеров) букв в выравнивании A отличается хотя бы одним номером от колонки букв в выравнивании B, то они выровнены не одинаково
3. Алгоритм сравнения

1) Для каждого выравнивания идём по колонкам N = 1,2,3,... и составляем вектор S:

N: SI(N) = s1,s2,...., s_n

s1 – номер буквы в первой последовательности

s2 – номер буквы во второй последовательности

.....

Если в последовательности i стоит гэп, то si = “-”

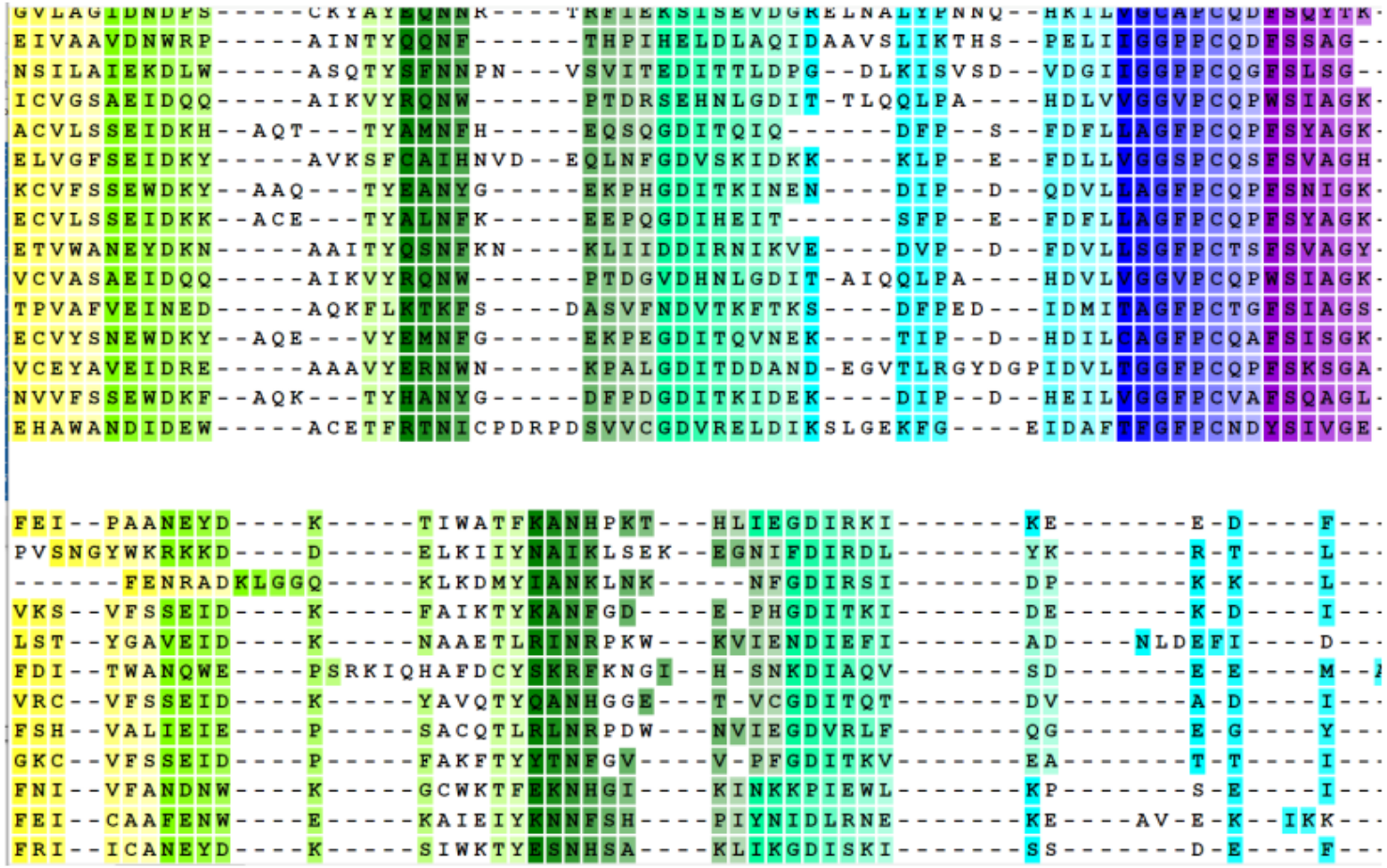
2) Если SI(N) = SII(N’), то колонка N выравнивания I выровнена одинаково с колонкой N’ выравнивания II

3) Последовательно идущие в обеих выравниваниях одинаково выровненные колонки объявляются блоком одинаково выровненных фрагментов

Готовые программы для сравнения выравниваний

- VerAlign: a multiple sequence alignment assessment tool (https://www.ibi.vu.nl/programs/veralignwww/)
- AlignStat: A tool for the statistical comparison of alternative multiple sequence alignments (https://github.com/TS404/AlignStat?tab=readme-ov-file)

VerAlign – программа для ручного сравнения – раскрашивает колонки второго выравнивание цветами первого



Практическое задание в классе

- Дано: два выравнивания pf00145_seed-reduced.fasta и pf00145_seed-tcoffee-reduced.fasta одних и тех же последовательностей
- Найти
  - ✓ блок одинаково выровненных колонок в двух выравниваниях
  - ✓ участок, на котором выравнивания полностью различны
  - ✓ результаты сохранить в гугл-форме

Участок несовпадающих выравниваний – между двумя совпадающими блоками