DNA (Евдокимов)

Описание программы

Задача программы состоит в выравнивании двух последовательностей, которое используется в биоинформатике при построении выравниваний аминокислотных или нуклеотидных последовательностей в Python.
Используется алгоритм Нидлмана-Вунша.
Для начала по заданным данным генерируются 2 последовательности ДНК. Затем высчитываются коэфициенты для подсчета матрицы схожести: d - штраф за разрыв равный длине самой длинной последовательности, A = коэфициент схожести символов равный квадрату штрафа за разрыв.
Затем непосредственно составляется матрица. В конце концов по конечной матрицы мы можем востановить необходимое нам выравнивание.

Тесты и эксперементы

(Можно самостоятельно запустить программу) Проверим программу на разных данных:

Тест 1

Вот наши последовательности
GCACT
GGTGA
Проводим выравнивание
—GCACT
GGTG-A–
Объединяем
GGTGCACT

Тест 2

Вот наши последовательности
TCGGAGGTTT
GTACAGCTTG
Проводим выравнивание
-T-CGGAG-GTT-T
GTAC–AGC-TTG-
Объединяем
GTACGGAGCGTTGT

Тест 3

Вот наши последовательности
TGAAAGGTATCGAATTCCCTAACTACGGTA
ACATTAGCCTTCCCGACCTCAAAAGGCCAAGTCCT
Проводим выравнивание
TGA-A–AG–GTAT–CGA–ATTC------CCTAA–CTACGGTA
–ACATTAGCC-T-TCCCGACC–TCAAAAGGCC-AAGTC–C--T-
Объединяем
TGACATTAGCCGTATCCCGACCATTCAAAAGGCCTAAGTCTACGGTA

Тест 4

Программа достаточно быстро работает даже на 1000 символах. Результат в папке с доп. файлами.

Тест 5

На 10000 символах программа работает заметно дольше, но тоже справляется. Под результат работы я выделил отдельный файл 10000.txt

Тест 6

Наконец тест на 10^5 символов. Программа очень долго работает. Результаты в файле.

Выводы

Смотря на результаты тестов, можно утверждать, что длина последовательности влияет на время работы. При том время растет нелинейно, а квадратично (для квадратной матрицы).