

Sprawozdanie z laboratorium 3

Wstęp do bioinformatyki

Piątek 09.15

Maciej Hołub 236518

15.05.2019

Link do githuba: <https://github.com/Holub0816/Bioinformatics/tree/Zadanie3>

1. Cel

Celem laboratorium było napisanie programu znajdującego optymalne lokalne dopasowanie pary sekwencji DNA. Przeprowadzenie jakościowej analizy złożoności obliczeniowej czasowej i pamięciowej dla kilku przykładów kodu. Porównanie przykładowych par sekwencji ewolucyjnie powiązanych i niepowiązanych.

2. Funkcjonalność programu

Program posiada możliwość wczytania sekwencji nukleotydów w formacie FASTA z pliku tekstowego, bezpośrednio z bazy danych NCBI lub wczytując kod z klawiatury. Sekwencje są parsowane w celu oddzielenia sekwencji od ich identyfikatora. Schemat punktacji substytucji i konwersji wprowadzany jest z pliku tekstowego mającego postać macierzy za pomocą funkcji `substMatrix()`. Następnie za pomocą funkcji `points()` tworzona jest macierz punktacji dopasowania wypełniana według algorytmu Smitha - Watermana oraz pomocnicza macierz pokazująca wszystkie lokalne ścieżki dopasowania (funkcja `generateHelpMatrix()`). Funkcja wyświetla macierz pomocniczą w formie graficznego wykresu przedstawiającego wszystkie ścieżki dopasowania. Funkcja `displayMatrix()` umożliwia wyświetlenie ścieżek dopasowań w jednym oknie graficznym, a zapisanie tego okna do pliku z rozszerzeniem .png jest możliwe za pomocą metody `saveMatrixToPngFile()`. Funkcja `equations()` wyświetla parametry programu i dopasowania dla wszystkich możliwych dopasowań (wynik, długość dopasowania, liczbę przerw, liczbę pasujących nukleotydów). Parametry te można zapisać do pliku tekstowego używając funkcji `saveToFile()`. Uzyskane sekwencje ze wstawionymi przerwami można zapisać do wybranego pliku tekstowego w formacie przypominającym format FASTA za pomocą funkcji `saveToFastFile()`. Program jest wywoływany z linii komend.

Przykładowe wywołanie (obie sekwencje wprowadzone ręcznie, za pomocą klawiatury):

```
interface('input1','GCTAGCA','input2','ACAGTAGC','gap',-1,'filename','myfile.txt');
```

Wynik działania programu:

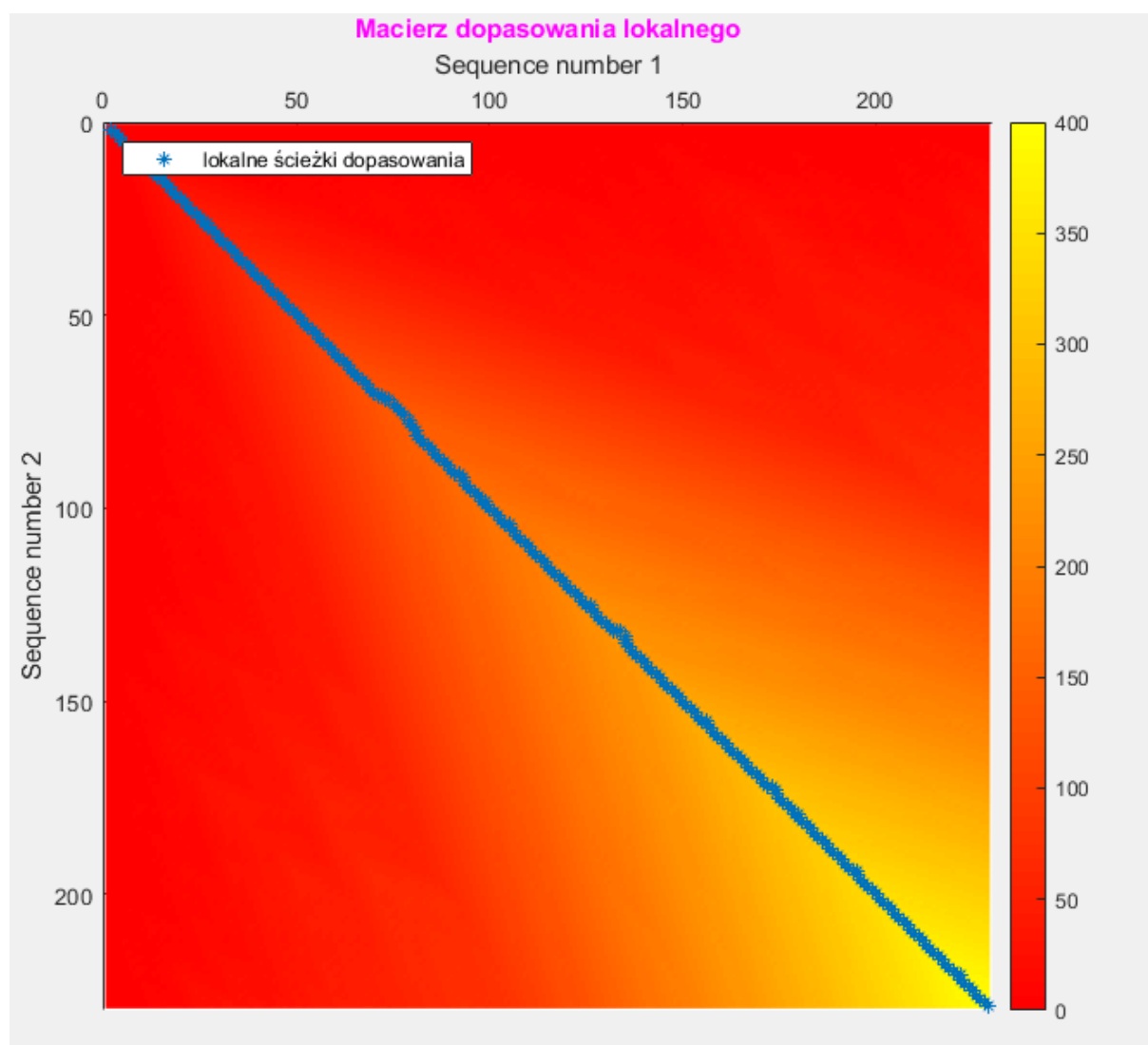
```
# seq1: -GCTAGCA
# seq2: -ACAGTAGC
# Mode: SIMILARITY
# Punctuation: AA  AC  AT  AG  CC  CT  CG  TT  TG  GG
#                2  -7  -5  -7  2  -7  -5  2  -7  2
# Gap: -1
# Score: 8
# Length: 7
# Gaps: 2/6 (33%)
# Identity: 3/6 (50%)
GCTAGA
|||
-GTAG-
```

3. Analiza złożoności obliczeniowej

Obliczone złożoności obliczeniowe i przestrzenne są podane na koniec każdej metody. Analiza została przeprowadzona dla wszystkich metod użytych w programie. W każdym z wypadków obliczona została złożoność dla największej możliwej ilości operacji. Dodatkowo szacuję złożoność obliczeniową korzystając z notacji dużego O. Dla funkcji points() generującej macierz dopasowania złożoność obliczeniowa wynosi : $O(n^2)$. Dla funkcji generateHelpMatrixx() złożoność obliczeniowa wynosi $O(n)$. Złożoność pamięciowa wynosi $O(2n^2)$ i wynika to z przechowywania w postaci tablic dwóch macierzy – macierzy punktacji oraz macierzy przedstawiającej ścieżkę optymalnego dopasowania.

4. Porównanie sekwencji

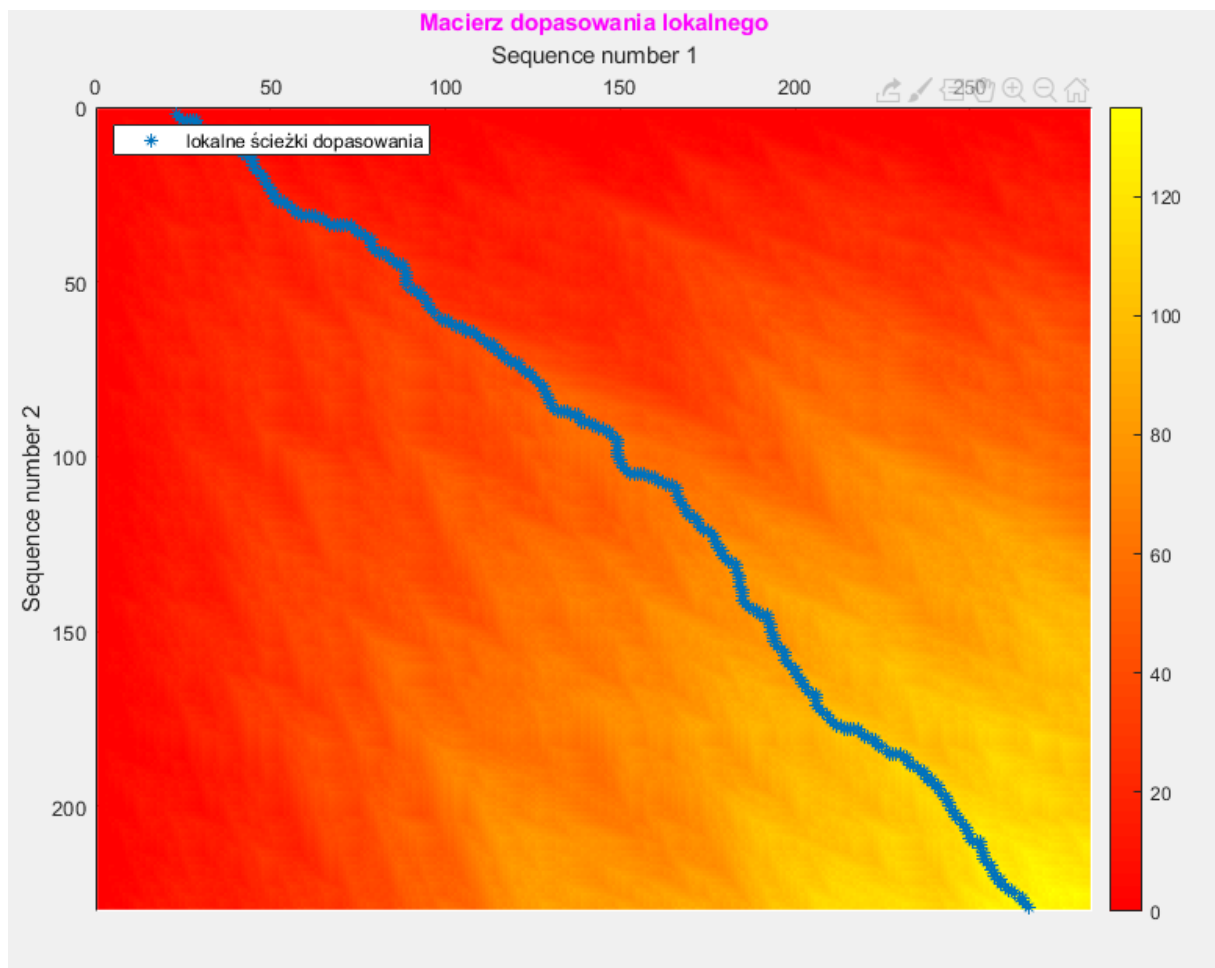
a) Porównanie sekwencji powiązanych ewolucyjnie – Stoń Azjatycki i Mamut – gen kodujący białko cytochrom b



1 lokalne dopasowanie:

```
# seq1: GAAATTTCGGCTCACTACTAGGAGCGTGCCTAATTACCCAAATCCTAACAGGATTATTCTAGCCATACATTACACACCTGACACAATAACTGCATTTTCATCCATATCCCATACTGCGGAGACGTCAACTACGGCTGAATTATTGCAAA
# seq2: GAAATTTCGGCTCACTACTAGGAGCATGCCTAATTACCCAAATCCTAACAGGATTATTCTAGCCATACATTATACACCCGACACAATAACCGCATTTCTCATCTATATCCCATACTGCGGAGATGTCAACAAATGGCTGAATTATTGCAAA
# Mode: SIMILARITY
# Punctuation: AA  AC  AT  AG  CC  CT  CG  TT  TG  GG
#               2   -7  -5   -7   2   -7   -5   2   -7   2
# Gap: -1
# Score: 400
# Length: 242
# Gaps: 12/234 (5%)
# Identity: 204/234 (87%)
GAAATTTCGGCTCACTACTAGGAGCGTGCCTAATTACCCAAATCCTAACAGGATTATTCTAGCCATAC-A-TTACACACCTGACACAATA-ACTGCATTTTCATCCATATCCCATACTGCGGAGACGTCAA--CTACGGCTGAATTATTGCAAA
|||||
GAAATTTCGGCTCACTACTAGGAGCATGCCTAATTACCCAAATCCTAACAGGATTATTCTAGCCATACATTATACA-CC-CGACACAATAAC-CGCATTCTCATCTATATCCCATACTGCGGAGATGTCAACAA--TGCGTGAATTATTGCAAA
```

b) Porównanie sekwencji niepowiązanych ewolucyjnie – cytochrom b Mamuta i białko STS dzika euroazjatyckiego

[illegible]

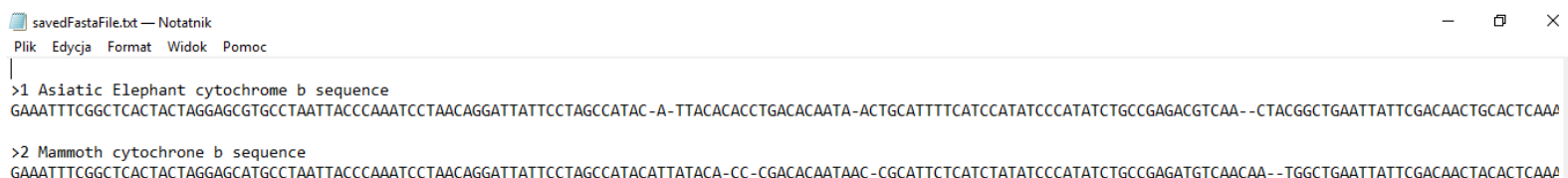
W przypadku sekwencji powiązanych ewolucyjnie, widać, że dopasowanie lokalne jest prawie identyczne jak powiązanie globalne (w zasadzie w tym przypadku można je tak traktować). Podobieństwo sekwencji wynosi 87 % i jest to również związane z większym kosztem niedopasowania nukleotydów sekwencji porównywanych niż z karą za przerwy. Na Rysunku 2 widać, że sekwencje nie są powiązane ewolucyjnie, ponieważ ścieżka dopasowania, mimo że tylko jedna, ma bardzo nieregularny przebieg. Podobieństwo sekwencji wynosi tylko 24 %.

5. Przykład zapisu sekwencji do pliku w formacie FASTA.

Poniżej zaprezentowany jest przykładowy skrypt zapisania pliku (w tym przypadku zapisywane są sekwencje pochodzące z punktu 5a i są zapisywane do pliku tekstowego 'savedFastaFile.txt').

```
Input informations to FASTA-like file:
identifier of sequence 1: 1
description of sequence 1: Asiatic Elephant cytochrome b sequence
identifier of sequence 2: 2
description of sequence 2: Mammoth cytochrome b sequence
```

A tu pokazany jest widok na zapisany plik tekstowy:



```
savedFastaFile.txt - Notatnik
Plik  Edycja  Format  Widok  Pomoc

>1 Asiatic Elephant cytochrome b sequence
GAAATTTTCGGCTCACTACTAGGAGCGTGCCTAATTACCCAAATCCTAACAGGATTATTCCTAGCCATAC-A-TTACACACCTGACACAATA-ACTGCATTTTCATCCATATCCCATATCTGCCGAGACGTCAA--CTACGGCTGAATTATTCGACAACTGCACTCAA

>2 Mammoth cytochrome b sequence
GAAATTTTCGGCTCACTACTAGGAGCATGCCTAATTACCCAAATCCTAACAGGATTATTCCTAGCCATACATTATACA-CC-CGACACAATAAC-CGCATTCTCATCTATATCCCATATCTGCCGAGATGTCAACAA--TGGCTGAATTATTCGACAACTACACTCAA
```

6. Schemat blokowy algorytmów użyty do wygenerowania optymalnej macierzy dopasowania

Schemat blokowy znajduje się w folderze pod nazwą `algorytm_sciezka_dopasowania.png`. Algorytm dotyczy funkcji `generateHelpMatrix()`, która tworzy macierz dopasowania.