# Wstęp do bioinformatyki

# Nr ćwiczenia: 3

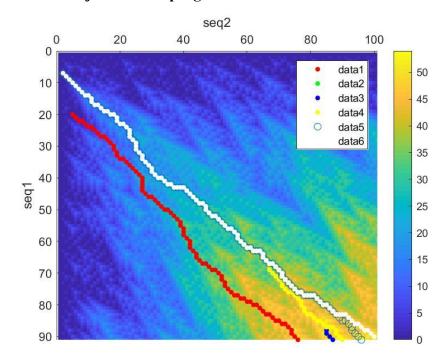
# Temat ćwiczenia: Dopasowanie lokalne par sekwencji

Nazwisko i Imię prowadzącego kurs: dr inż. Witold Dyrka

Wykonawcy:	
Imię i Nazwisko	Edyta Krukowska 217097, WPPT
Nr indeksu, wydział	
Termin zajęć: dzień tygodnia, godzina	Piątek 11.15
Data oddania sprawozdania	15.02.2019

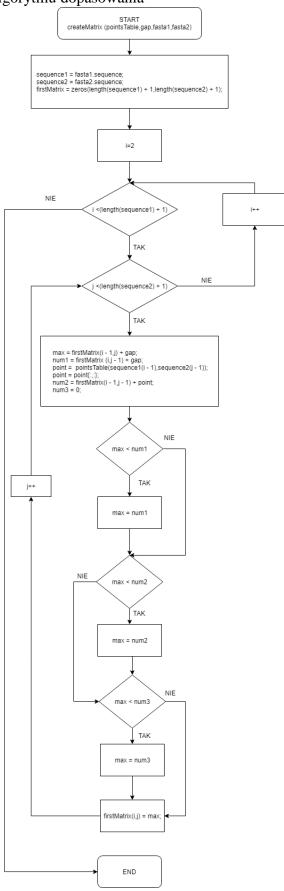
Repozytorium: <a href="https://github.com/Edie1995/Bioinformatyka/tree/zadanie3">https://github.com/Edie1995/Bioinformatyka/tree/zadanie3</a>

# 1. Prezentacja działania programu:

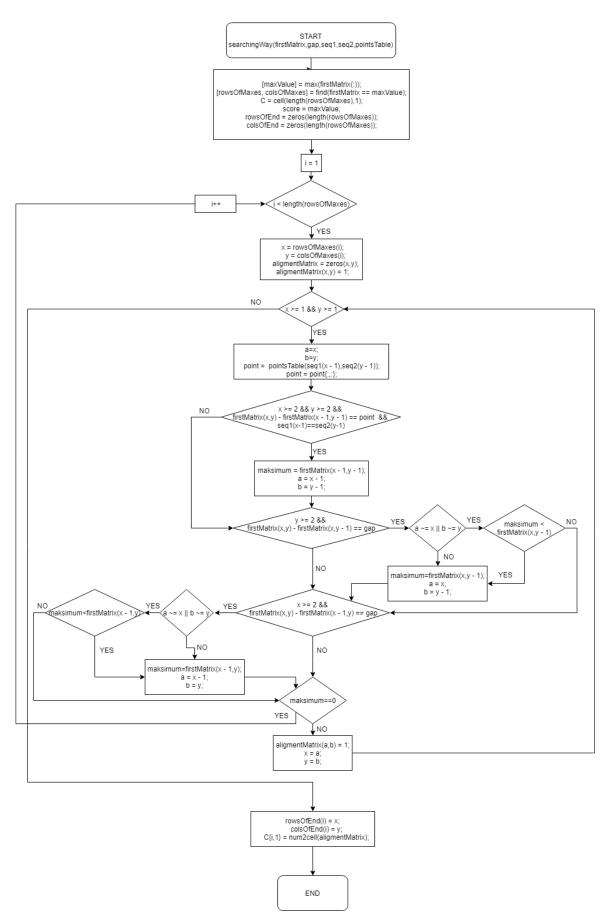


Rysunek 1 Przykład działania programu, dla przykładowych sekwencji wpisanych z klawiatury

# 2. Schematy blokowe algorytmu dopasowania



Rysunek 2 Schemat blokowy -tworzenie macierzy kosztu



Rysunek 3 Schemat blokowy -tworzenie lokalnej najoptymalniejszej drogi

# 3. Oszacowanie złożoności czasowej obliczeniowej i pamięciowej kodu poszczególnych funkcji i całego programu:

#### > Czasowa:

#### createMatrix:

3 operacje if + 4 przypisania wartości \* (m)(n) = O(mn);

#### searchingWay:

Przyjmując pesymistyczny wariant szukania maksimów = (m)\*(n)\*(n)

Funkcja while w pesymistycznej wersji obejmie (m)\*(n)\*( 3 porównania i 6 przypisań wartości)

Podsumowując, w negatywnym wariancie cała funkcja (c > 0) c\*(m)(n) więc jest ona maksymalnie rzędu O(mnn)

#### saveToFile:

W pesymistycznym założeniu, lecz raczej nie możliwym do osiągnięcia, że ścieżka będzie szła wzdłuż krawędzi macierzy jest to funkcja rzędu O(m+n)

#### writeSequence:

W funkcji while według założenia pesymistycznego otrzymamy m\*n\*(6 porównań (maksymalnie) i 2 przypisania. Jest to największy while w tej funkcji, pozostałe są rzędu o(mn), czyli na pewno niższego niż O(mn). Z takiego oszacowania wynika, że maksymalny tej funkcji to O(mn).

Pozostałe funkcje wykorzystywane przez program są rzędu stałego O(1) lub liniowego O(n) lub O(m). Na tej podstawie można oszacować, ze dla cały program jest maksymalnie rzędu O(mnn).

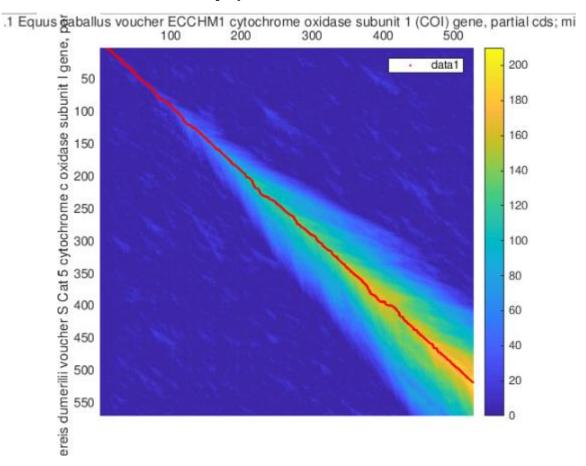
## > Pamięciowa

Największym obiektem stworzonym w całym programie jest macierz kosztu i jest ona rzędu O(mnn), wszystkie pozostałe obiekty wykorzystywane w programie są rzędu takiego samego, lub niższego o(mn)<O(mn). Możemy więc uznać, że jest to złożoność pamięciowa  $c^*m^*n$  (gdzie c > 0), czyli O(mn).

# 4. Porównanie przykładowych par sekwencji

# 4.1. Ewolucyjnie powiązanych

• Porównanie sekwencji cytochromu C kota i konia

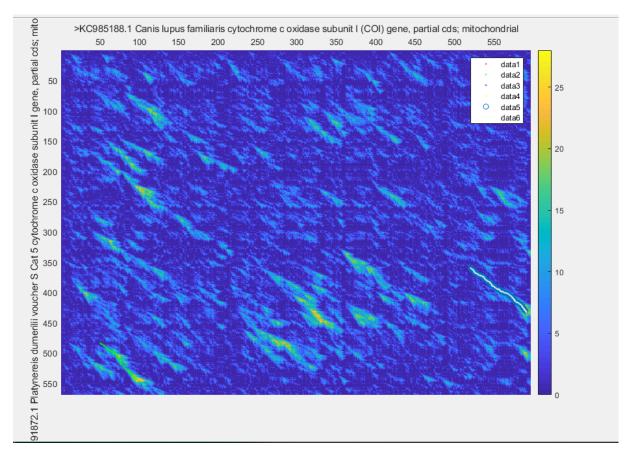


Rysunek 4 Macierz kosztu z naniesionymi ścieżkami lokalnego dopasowania porównująca cytochrom C kota (oś y) i konia (oś x)

```
#1: XKC591872.1 Platymereis dumerilii voucher S Cat 5 cytochrome c oxidase subunit I gene, partial cds; mitochondrial
#2: XJQ735459.1 Equus caballus voucher ECCHM1 cytochrome oxidase subunit 1 (COI) gene, partial cds; mitochondrial
#Mode: distance
#Table:
# A C 6 T
A 2-7-5-7
C-7-2-7-5
G-5-7 2-7
T-7-5-7 2
#Gasp: -2
#Score: 210
#Length: 655
#Indentity: 259/655(42%)
#Gaps: 275/655(42%)
```

Rysunek 5 Konsolowy wynik działania programu dla cytochromu C kota i konia

# • Porównanie cytochromu c kota z psem



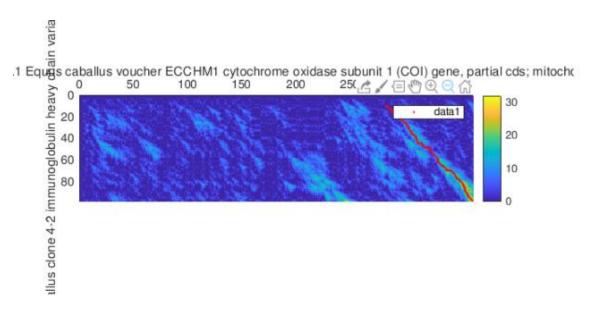
Rysunek 6 Macierz kosztu z naniesionymi ścieżkami lokalnego dopasowania porównująca kota (oś y) z psem (oś x)

```
#1: >KC591872.1 Platynereis dumerilii voucher S Cat 5 cytochrome c oxidase subunit I gene, partial cds; mitochondrial
#2: >KC985188.1 Canis lupus familiaris cytochrome c oxidase subunit I (COI) gene, partial cds; mitochondrial
#Mode: distance
#Table:
# A C G T
A 2-7-5-7
C-7 2-7-5
G-5-7 2-7
T-7-5-7 2
#Gap: -2
#Score: 28
#Length: 86
#Indentity: 28/86(33%)
#Gaps: 36/86(42%)
T-TTATCAC-AA-C-AGTCA-T-CAAT-ATACGATC-AAAAGGG-C--T--AA--A-AT-TAGAACGCGTTC-C-TTTATTTGTAT
             TTCT-A-CCAT-AGAA--GATGCTAATAGT-AG--AAGAAAG--GATGGAGGAAGGAG-TCAGA-AG-C-T-CATGTTATT--TAT
```

Rysunek 7 Konsolowy wynik działania programu dla Kota i Psa

## 4.2. Ewolucyjnie niepowiązanych

## • Porównanie cytochromu c i b konia



Rysunek 8 Macierz kosztu z naniesionymi ścieżkami optymalnego dopasowania wygenerowana dla łańcucha immunoglobuliny (oś y) oraz cytochromu c (oś x) konia

```
#1: >DQ125446.1 Equus caballus clone 4-2 immunoglobulin heavy chain variable region mRNA, partial cds
#2: >JQ735459.1 Equus caballus voucher ECCHM1 cytochrome oxidase subunit 1 (COI) gene, partial cds; mitochondrial
#Mode: distance
#Table:
# A C G T
A 2-7-5-7
C-7 2-7-5
G-5-7 2-7
T-7-5-7 2
#Gap: -2
#Score: 32
#Length: 108
#Indentity: 39/108(36%)
#Gaps: 46/108(43%)
A-ACTGAA-G-GAG-TCAGG-ACCTGGCCTGG-TGAAGCCC---TCGC----AGACTCTCTCCCTCACC-TGCG----CTGTCTCT-GGATTCTCTTTGAGAGATGCC
              -111
                                                                                 Ш
1 11
                                                                                           1
                                                                                               1111
AATCT--GGCGC-ATGCAGGAGC-C-G-CT-GTTG--A--CTTAA-CCATTTT--GTCTCTC---CACCTAGC-GGGGGTGT-C-CGCGAT-T-T---TA-G-GTGCC
```

Rysunek 9 Konsolowy wynik działania programu dla cytochromu b (oś y) oraz cytochromu c (oś x) konia