Sprawozdanie z laboratorium 4

Wstęp do bioinformatyki

Piątek 09.15
Maciej Hołub 236518
29.05.2019
Link do githuba: https://github.com/Holub0816/Bioinformatics/tree/Zadanie4_popr

1. Cel

Celem laboratorium było napisanie programu znajdującego dopasowanie wielu par sekwencji DNA. Przeprowadzenie jakościowej analizy złożoności obliczeniowej czasowej i pamięciowej dla kilku przykładów kodu. Porównanie kilku par sekwencji ewolucyjnie. Wygenerowanie ogólnego schematu blokowego działania aplikacji.

2. Funkcjonalność programu

Program posiada możliwość wczytania paru sekwencji nukleotydów lub białek w formacie FASTA z pliku tekstowego lub bezpośrednio z bazy danych NCBI. Sekwencje są pobierane z pliku lub z bazy za pomocą funkcji loadSequencesFromFile() oraz parsowane parsowane w celu oddzielenia sekwencji od ich identyfikatora (metoda parseFasta()).
Schemat punktacji substytucji i konwersji wprowadzany jest z pliku tekstowego mającego postać macierzy za pomocą funkcji substMatrix(). Sekwencja centralna, do której będzie można dopasowywać inne sekwencje za pomocą algorytmu gwiazdy jest znajdowana za pomocą funkcji findCentralSequence() korzystające z funkcji distanceMatrix() (to ona oblicza koszty dopasowań poszczególnych par sekwencji). Następnie za pomocą funkcji matrix() tworzona jest macierz punktacji dopasowania wypełniana według algorytmu Needlemanna-Wunscha oraz pomocnicza macierz pokazująca wszystkie lokalne ścieżki dopasowania (funkcja generateHelpMatrix()). Później za pomocą funkcji alignmentSequence() dopasowuje wszystkie sekwencje do sekwencji centralnej pilnując, aby wstawiać przerwy w miejsca, gdzie wstawione zostały one w sekwencji centralnej. Za pomocą funkcji calculateResult() obliczam całkowity koszt dopasowania, a wyświetlenie sekwencji po dopasowaniu zgodnie z formatem CLUSTAL OMEGA umożliwia funkcja displaySequences(). Zapisanie sekwencji w formacie FASTA ma miejsce za pomocą funkcji saveToFile(). Program jest wywoływany z linii komend.

Przykładowe wywołanie porównujące ze sobą sekwencje z Wykładu ze wstępu do bioinformatyki :

fastaFile.txt:

```
>FirstSequence
GCACAT
>SecondSequence
TGAGA
>ThirdSequence
GACA
>FourthSequence
GAACT
```

testScript.m:

interface('filename1','fastaFile.txt','filename2','myfile.txt')

Wynik widoczny w linii poleceń:

```
>FirstSequence                          G-ACA-  4
>SecondSequence                         GCACAT  6
>ThirdSequence                          TGAGA-  5
>FourthSequence                         GAACT-  5
                                          *
Całkowity koszt dopasowania wielu sekwencji wynosi 24.
```

3. Porównanie sekwencji

Porównanie sekwencji powiązanych ewolucyjnie – Słoń Indyjski, Słoń Afrykański, Słoń Leśny i Mamut – fragment genu kodującego białko cytochrom b

```
>U23740.1                               GAAATTTCGGCTCACTACTAGGAGCGTGCCTAATTACCCAAATCCTAACAGGATTATTCCTAGCCA  66
>U23741.1                               GAAATTTCGGCTCACTACTAGGAGCATGCCTAATTACCCAAATCCTAACAGGATTATTCCTAGCCA  66
>AY424307.1                             GAAATTTCGGCTCACTACTAGGAGCGTGCCTAATTACCCAAATCCTAACAGGATTATTCCTAGCCA  66
>NorthAmericanMastodon                  GAAATTTCGGCTCACTACTAGGAGCATGCCTAATTACCCAAATCCTAACAGGATTATTCCTAGCCA  66
                                        ************************ ************************************************
Całkowity koszt dopasowania wielu sekwencji wynosi 4.
```

Jak można zobaczyć na załączonym rysunku prawie wszystkie kolumny uległy konserwacji. Świadczy to o ogromnym podobieństwu porównywanych sekwencji.

4. Analiza obliczeniowa

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(n2).
Dla funkcji generateHelpMatrixx() złożoność obliczeniowa wynosi O(n).
Złożoność pamięciowa wynosi O(x * 2n2) i wynika to z przechowywania w postaci tablic dwóch macierzy – macierzy punktacji oraz macierzy przedstawiającej ścieżkę optymalnego dopasowania i liczbie wykonywaniu się tych metod w zależności od liczby wprowadzonych sekwencji.

5. Schematy blokowe algorytmów znajdują się w folderze i mają rozszerzenie .png