Bioinformatyka strukturalna

Autor: Anna Glinka
Temat: Porównanie zbioru struktur drugorzędowych RNA (z pominięciem pseudowęzłów)

Wstęp

Drugorzędowe struktury RNA ściśle wiążą się z funkcją którą pełnią w żywych organizmach. Dlatego też ewolucyjnie konserwowane wydają się być nie tylko same sekwencje RNA, ale także struktury wyższych rzędów [1]. W ciągu ostatnich lat stworzono wiele algorytmów służących porównywaniu struktur RNA, wydaje się jednak, że do dziś nie stworzono jednego do końca zadowalającego algorytmu [4]. W ramach projektu zaimplementowano więc 3 różne podejścia do prezentowanego problemu: porównanie za pomocą metryki Hausdorffa, metryki „góry" oraz metryki „drzewa". Każda z metod daje inne wyniki i zwraca uwagę na inne elementy struktury, dlatego też nie jest możliwe bezpośrednie porównywanie wyników uzyskanych różnymi metodami.

Cel

Celem prezentowanego projektu jest implementacja algorytmów umożliwiających porównanie zbioru struktur drugorzędowych RNA nie zawierających pseudowęzłów, przedstawionych w notacji dot-brackets.

Metodyka

Metryka „drzewa"

Metoda należy, do tzw. metod gruboziarnistych, w których nie analizujemy poszczególnych par, a jedynie całe struktury drugorzędowe. Do stworzenia drzewa wymagane jest najpierw przyporządkowanie do sekwencji nawiasów i kropek odpowiednich struktur drugorzędowych. Z ciągu rozpoznanych struktur budowane jest następnie drzewo, w którym wierzchołki odpowiadają rozpoznanym częściom, jak na przykład pętla, spinka do włosów czy wybrzuszenie, a krawędzie to fragmenty sparowanych nukleotydów pnia. Wierzchołkom przyporządkowane zostają numery zgodnie z liczbą odchodzących od nich krawędzi (pętla, wybrzuszenie, niesparowane fragmenty na końcach sekwencji = 1, spinka do włosów = 0, skrzyżowanie – zależy od liczby odchodzących gałęzi). Dalsza część algorytmu obejmuje przeszukiwanie drzew porównywanych sekwencji w głąb. Stworzone w ten sposób ciągi liczb są następnie dopasowywane (w prezentowanym projekcie wykorzystano algorytmu Needleman -Wunsh'a), co daje wzajemne podobieństwo sekwencji [3].

Zastosowane skróty i oznaczenia :
d – pień
s – spinka do włosów
l – niesparowane nukleotydy na końcach łańcucha
p – pętla wewnętrzna
w – wybrzuszenie
x – skrzyżowanie

Szczegóły implementacyjne – metryka „drzewa"

Funkcja elementyStruktury

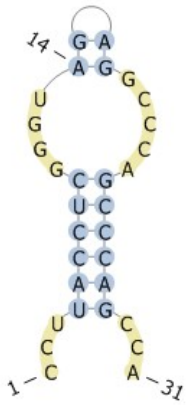
Do rozpoznawania struktur wykorzystano strukturę stosu (odnalezienie łączących się nukleotydów) oraz wyrażenia regularne. Stos dodatkowo pełni funkcję kontrolną, w przypadku podania nieprawidłowej liczby nawiasów zgłoszony zostaje błąd. Program umożliwia analizę struktur posiadających pseudowęzły, w takim wypadku nawiasy inne niż „(", „)" zostają zamienione na kropki oznaczające brak parowania. Parametrem przekazywanym do funkcji jest sekwencja w notacji kropkowo-nawiasowej, wartością zwracaną jest natomiast stri2 (ciąg symboli będący

reprezentacją obecnych w RNA struktur drugorzędowych oraz ich topologii, poszczególne gałęzie drzewa oddzielane są nawiasami „]”, „[”).
Wyłącznie dla potrzeb implantacyjnych zastosowano dodatkowe oznaczenie:
r – oznaczenie struktury prezentowanej w notacji kropkowo-nawiasowej podobnie do spinki do włosów, jednak bez niesparowanych nukleotydów w środku. np. ((()))
z – oznaczenie fragmentu z lewej strony będącego odpowiednikiem wybrzuszenia występującego po prawej stronie struktury, np. ((((((...((((…..))))…)))…))). Dodatkowo wyświetlane jest rozpoznanie struktury przyporządkowane do konkretnego nukleotydu – stri.

Funkcja drzewoDFS
Argumentem przekazywanym funkcji jest stri2 (kod drzewa zwracany przez poprzednią funkcję), z którego odczytuje ona ciąg cyfr, odpowiadający przeszukiwaniu grafu w głąb. Zwracana sekwencja wykorzystywana jest następnie przez funkcję NeedlemanWunsch.

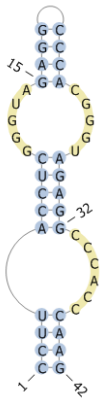
Funkcja NeedlemanWunsch
Funkcja przyjmuje 2 parametry, będące sekwencją powstałą w czasie czytania porównywanych drzew algorytmem DFS. Punktacja podstawowa ustawiona została na dopasowywane = 1, niedopasowanie i przerwa = -1. Wyświetlane są sekwencje po dopasowaniu, procent ich identyczności oraz punktacja.

Testowanie rozpoznawania struktury
Struktury stworzone na potrzeby testowania:
Sek 1
CCUUACCUCGGGUAGAGGCCCAGCCCAGCCA
...((((((....(()).....))))))…




```
RNA:   ...((((((....(()).....))))))...
stri:  lllddddddppppddddpppppddddddlll
stri2:  lprpl
dfs:  110
```

Sek 2
CCUUACCUCGGGUAGAGGCCCACGGGUAGAGGCCCACCCAAG
(((((((((.....((((()))).....)))))..…))))

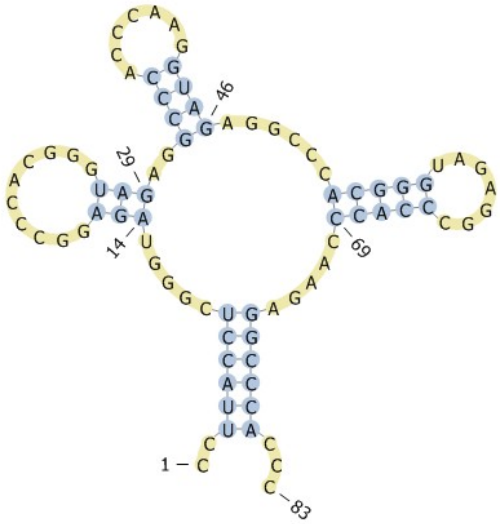


```
RNA:   (((((((((.....(((()))))).....)))))......))))
stri:  ddddddddddpppppddddddddpppppddddddwwwwwwddddd
stri2:  zprpw
dfs:  110
```

Sek 3

CCUUACCUCGGGUAGAGGCCCACGGGUAGAGGCCCACCCAAGGUAGAGGCCCACGGG
UAGAGGCCCACCCAAGAGGCCCACCC

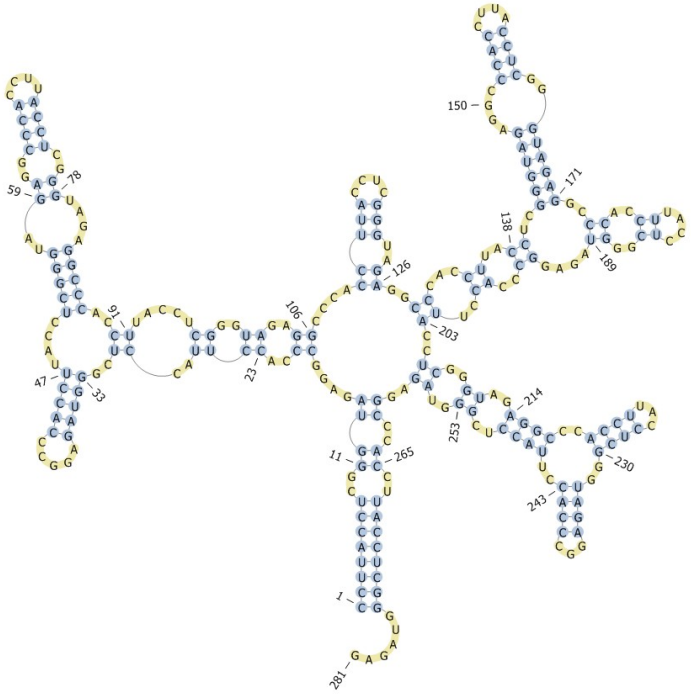
..((((((.....(((..........))).((((.......)))).....(((((.......)))))).....))))))…



```
RNA:   ..((((((.....(((..........))).((((.......)))).....(((((.......)))))).....)))))...
stri:  llddddddxxxxxddddsssssssssssdddxxddddsssssssddddxxxxxxddddddsssssssddddddxxxxxddddddlll
stri2:  l[s][[s]][s]l
dfs:  13000
```

Sek 1_1

CCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGU
AGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUU
ACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAG
GCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCU
CGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAG

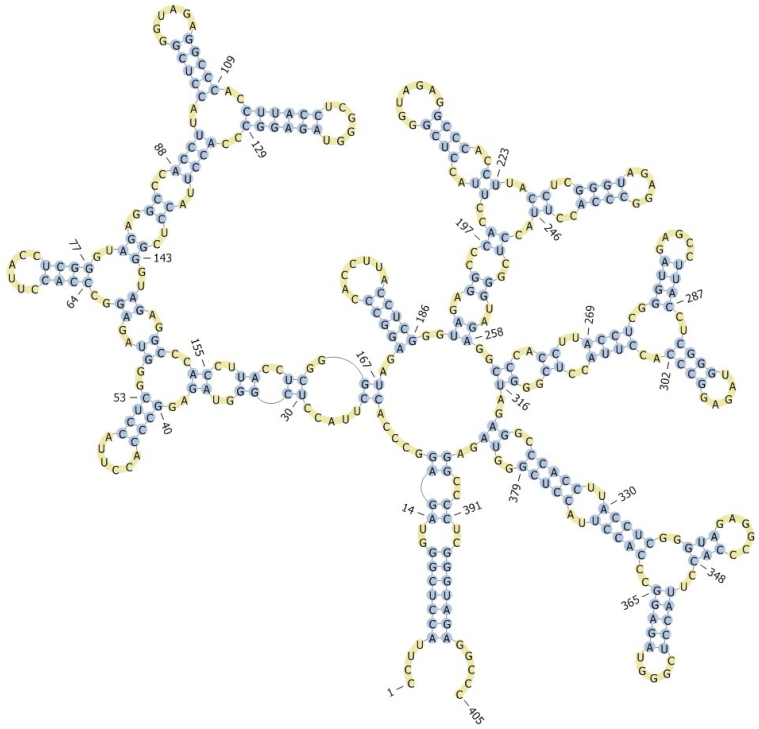


(((((((..((((....((..((((..((..(((((.....)))))....((((..((..((((....))))..))....)))))..))....))..))..))....(((((....)))..))..((...
((..((..((((((....((((....))))..))))))..((..((....))..))....))..))..))..((..((..((((..((((...))))..
((((...))))..))))..))..))..))..))..)))))))..…

```
pz[pzp[s][ppspp]pwp][zsw][pp[psp][psp]pp][pp[s][s]pp]wpl
[1, 1, 1, 4, 1, 1, 2, 0, 0, 1, 1, 2, 1, 0, 1, 0, 1, 0, 1, 1, 1, 2, 1, 1, 0, 0]
RNA:   (((((((..((((....((..((((..((..(((((.....)))))....((((..((..
((((....))))..))....))))))..))....))..))..))....(((((....)))..))..((...((..((..((((((....
((((....))))..))))))..((..((....))..))....))..))..))..((..((..((((..((((...))))..
((((...))))..))))..))..))..))..))..)))))))).....
stri:
ddddddddppddddxxxxddppddddppddxxddddddsssssddddddxxxxddddddppddppddddddssssddddppddppppddddddxxddpppppddwwddppddxxxxddddddssssdddwwddxxddpppddppddxxddddddddppppddddddssssddddppddddddxxddppddddssssddppddxxxxxddppddppddxxddppddppddddxxddddddssddddxxddddddssddddxxddddddppddppddxxddwwddppddddddddddllll
stri2:  pz[pzp[s][ppspp]pwp][[zsw]][[pp[psp][psp]pp]][pp[s][s]pp]wpl
dfs:  1114112001121010101112110 0
```

Sek 1_2

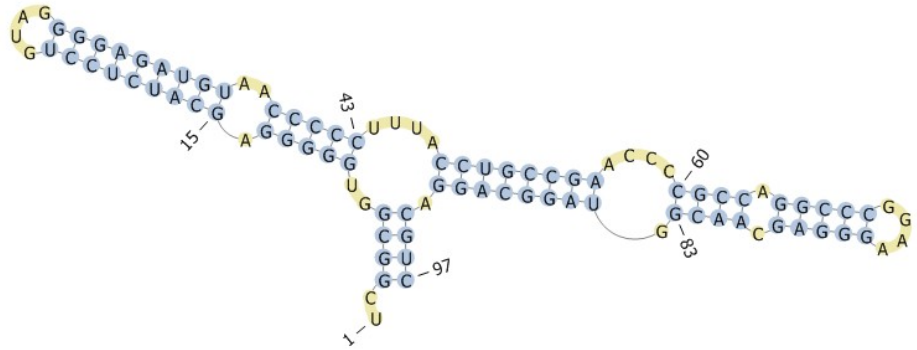
CCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGU
AGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUU
ACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAG
GCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCU
CGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGCCUU
ACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAG
GCCCACCUUACCUCGGGUAGAGGCCCACCUUACCUCGGGUAGAGGCCCCUCGGGUAG
AGGCCC



....(((((((..((((.....((.....((((..((..((((......))))..((..((..((((......))))..((..((..((((..(((((......)))))..
((((((......))))))..))))..))..))..))..))..))..))..))..))..((((((.......))))))..((..((..((..((..((((.......))))..))..))..
((((.....))))..))..))..))..))..((..((..((((..((((.....))))..((((.....))))..))))..))..))..((..(((((..(((((..(((......)))..
(((((.....)))))..))))))..))))))..))..))..)))))))).....

```
RNA:    ....(((((((..((((.....((.....((((..((..((((......))))..((..((..((((.......))))..((..((..((((..(((((......)))))..((..((..
((((..(((((......))))))..((((((......))))))..))))..))..))..))..))..))..))..))..))..))..))..))..))..))..((..
((((.......))))))..((..((..((..((..((((.......))))..))..))..((..((((.....))))..))..))..))..))..))..))..))..((..
((..(((((..((((.....))))..((((.....))))..))))..))..))..((..((((((..((((((..(((......)))..
((((((.....))))))..))))))..))))))..))..))..))..)))))))).....
stri:
llllddddddddppddddxxxxxddpppppddddppddxxddddsssssssddddxxddppddxxddddsssssssddddxxddppddppddddxxddddd
dsssssssddddddxxddddddsssssssddddddxxddddppddppddxxddppddxxddppddwwddppddxxddddddsssssssddddddxxddppdd
ppddxxddppddddsssssssddddppddxxddppddddsssssddddppddxxddppddppddxxddppddppddddxxddddsssssddddxxdd
ddsssssddddxxddddppddppddxxddppddddddppddddddxxddddsssssssddddxxddddddsssssddddddxxddddddppddddddppddxxddw
wddppddddddddllllll
stri2:  lpz[pzp[s][p[s][pp[s][s]pp]p]pwp][[s]][[pp[psp][psp]pp]][[pp[s][s]pp]][pp[s][s]pp]wpl
111511200112001121010011121211200000
```

Sekwencje i struktury z bazy RNAStrand:
Sek PDB_00547
UCGGCGGUGGGGGAGCAUCUCCUGUAGGGGAGAUGUAACCCCCUUUACCU
GCCGAACCCCGCCAGGCCCGGAAGGGAGCAACGGUAGGCAGGACGUC

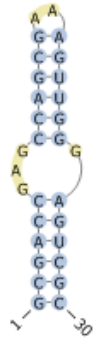


..((((..(((((.(((((((((....)))))))))..)))))....(((

(((((....((((.(((((....))))).)))).)))))))).))))

```
RNA:    ..((((..(((((.(((((((((....)))))))))..))))))....(((((((....((((.
((((((....)))))).)))).)))))))).))))
stri:
llddddxxddddddpddddddddddssssddddddddddppddddddxxxxddddddddddppppddddpddddddssssddddddpddddpddddddddddxddddd
stri2:  l[psp][ppspp]
dfs:  1211010
```

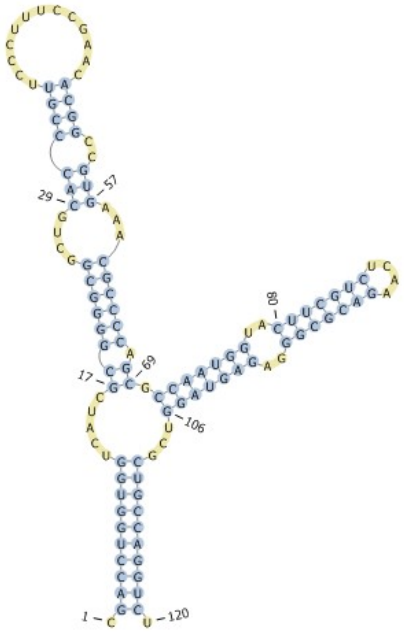
Sek PDB_00142
GCGACCGAGCCAGCGAAAGUUGGGAGUCGC



((((((...((((((..))))))..))))))

```
RNA:   ((((((...((((((..))))))..))))))
stri:  ddddddpppddddddssddddddpdddddd
stri2:  psp
dfs:  10
```

Sek CRW_00552
CGACCUGGUGGUCAUCGCGGGGCGGCUGCACCCGUUCCCUUUCCGAACAC
GGCCGUGAAACGCCCCAGCGCCAAUGGUACUUCGUCUCAAGACGCGGGAG
AGUAGGUCGCUGCCAGGUCU

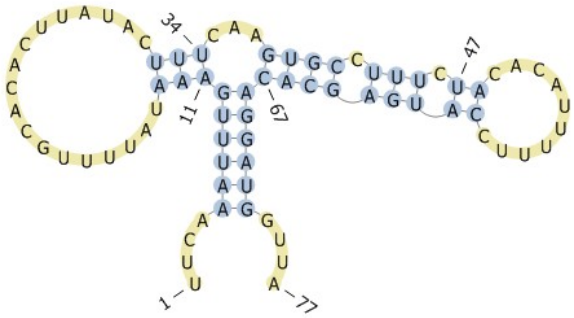


.((((((((((.....((((((((....(((((((.............))
))..)))...))))))).)).(((((((..(((((((....)))))))..)
))))))...))))))))))).

```
RNA: .((((((((((.....((((((((....(((((((.............))))..)))...))))))).)).(((((((..
(((((((....)))))))..)))))))...))))))))))).
stri:lddddddddddxxxxxddddddddpppppddddddddsssssssssssssddddwwdddpppddddddwddxddddddddppddddddddssssdd
ddddddppddddddddxxxddddddddddl
stri2:  l[zpzswpw][psp]]l
dfs:  12101110
```

Sek RFA_00642

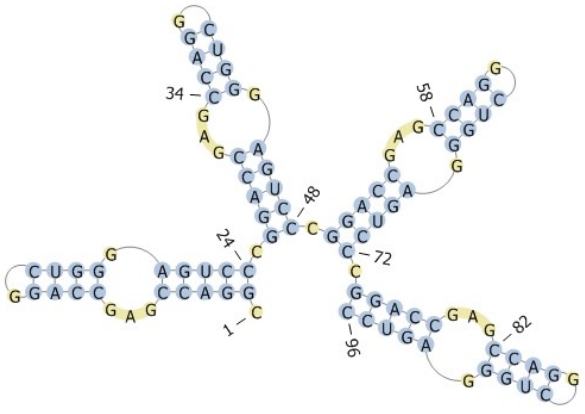
UUCAAAUUUGAAAUAUUUUGCACACUUAUACUUUCAAGUGCCUUUCUACA
CAUUUUUCCAUGAGCACAGGAUGGUUA
....(((((((((.................)))...((((.(((.((..
.......)))))))))))))....



```
RNA:   ....(((((((((.................)))...((((.(((.((.........))))))))))))))....
stri:  lllldddddddddsssssssssssssssssssddxxxddddwdddwddsssssssssssddddddddddddddddllll
stri2:  l[s][wws]l
dfs:  121100
```

Sek PDB_00176

CGGACCGAGCCAGGCUGGGAGUCCCGGACCGAGCCAGGCUGGGAGUCCCG
GACCGAGCCAGGCUGGGAGUCCCGGACCGAGCCAGGCUGGGAGUCC
.(((((...((((.)))).)))))..(((((...((((.)))).)))))..(
((((...((((.)))).)))))..(((((...((((.)))).)))))



```
RNA: .(((((...((((.)))).)))))..(((((...((((.)))).)))))..(((((...((((.)))).)))))..(((((...
((((.)))).)))))
stri:lddddddpppddddsddddpddddddxddddddpppddddsddddpddddddxddddddpppddddsddddpddddddxddddddpppddddsddddpd
ddddd
stri2:  l[psp][[psp]][[psp]][psp]
dfs:  1410101010
```

Metryka Hausdorffa

Metryka Hausdorffa umożliwia określenie odległości pomiędzy podzbiorami. W przypadku RNA została wykorzystana do porównywania zestawów par w poszczególnych sekwencjach. Odległość dh jest definiowana w 3 krokach: maksimum z odległości pomiędzy parami w porównywanych sekwencjach (Tabela 1), minimum z obliczonych poprzednio odległości po rzędach i kolumnach (Tabela 2), maksimum z tych minimalnych odległości i na końcu maksymalna wartość z tych dwóch [4].

Przykład 1.
S1 = ...((…)).
S2 = ..((…))..

PS1 = {{4, 10}, {5, 9}}
PS2 = {{3, 9}, {4, 8}}

Tabela 1.

Pary PS2/PS1	PS2_1	PS2_2
PS1_1	max(1, 1)	max(0 ,2)
PS1_2	max(2,0)	max(1, 1)

Tabela 2.

Pary PS2/PS1	PS2_1	PS2_2	Rz min
PS1_1	1	2	1
PS1_2	2	1	1
kol min	1	1	

da(BS1,BS2) – max z min po kolumnach =1
da(BS2,BS1) – max z min po rzędach = 1
dh – max z max = 1

Przykład 2.
S3 = ((((...))))..……
S4 = ((((...))))..........(...)…

PS3= {{1, 11}, {2, 10}, {3, 9}, {4, 8}}
PS4 = {{1, 11}, {2, 10}, {3, 9}, {4, 8}, {23, 27}}

Tabela 3.

Pary PS4/PS3	PS4_1	PS4_2	PS4_3	PS4_4	PS4_5	Rz min
PS3_1	0	1	2	3	22	0
PS3_2	1	0	1	2	21	0
PS3_3	2	2	0	1	20	0
PS3_4	3	2	1	0	19	0
kol min	0	0	0	0	19	

dh(S3, S4) = 19

Szczegóły implementacyjne - metryka Hausdorffa

Funkcja metrykaHausdorffa

Argumentami funkcji są sekwencje w notacji kropkowo-nawiasowej. Wykorzystuje ona strukturę stosu do identyfikacji par nukleotydów oraz dwuwymiarową macierz do dalszych obliczeń. Zwracaną wartością jest odległość porównywanych sekwencji.

Metryka „góry" (mountain metric)

Dla każdej sekwencji tworzony jest wektor z elementów będących różnicą pomiędzy liczbą nawiasów otwierających i zamykających. Poszczególne wektory są sumowane, a wartość bezwzględna pomiędzy różnicą stanowi odległość między sekwencjami [2, 4].

S1=..(((.....))).. vS1 = (0; 0; 1; 2; 3; 3; 3; 3; 3; 3; 2; 1; 0; 0; 0) = 24

S2=.((((.....)))). VS1 = (0; 1; 2; 3; 4; 4; 4; 4; 4; 4; 3; 2; 1; 0; 0) = 36

S3=..((((...)))).. vS3 = (0; 0; 1; 2; 3; 4; 4; 4; 4; 3; 2; 1; 0; 0; 0) = 28

dm(S1; S2) = |24 – 36| = 12 dm(S1; S3) = |24 – 28| = 4

Szczegóły implementacyjne – metryka „góry"

Funkcja metrykaGóry

Funkcja przyjmuje 2 argumenty będące sekwencjami w notacji kropkowo-nawiasowej. W implementacji również obecna jest struktura stosu. Funkcja zwraca wartość odległości pomiędzy sekwencjami.

Plik Bioinformatyka_projekt_wszystkie_funkcje.py zawiera pełną funkcjonalność programu, do poszczególnych funkcji została dodana obsługa plików, przyjmują one dodatkowe parametry, jak wejście lub indeks, umożliwiający zapisywanie wyników do wybranych plików oraz identyfikacje numeru sekwencji.

W folderze Testy znajdują się przykładowe testy projekty, natomiast w folderze Przykladowe_wyniki zaprezentowane są 2 różne pliki wynikowe.

Uwaga przed uruchomieniem:

W pliku Bioinformatyka_projekt_wszystkie_funkcje.py należy podać miejsce, gdzie znajdują się pliki z sekwencjami (sciezka) oraz gdzie chcemy mieć zapisany plik wynikowy (wyjscie).

Bibliografia:

1. GRUBER, ANDREAS R, BERNHART, STEPHAN H, HOFACKER, IVO L, & WASHIETL, STEFAN. (2008). *Strategies for measuring evolutionary conservation of RNA secondary structures*. BioMed Central Ltd. BioMed Central Ltd. http://www.biomedcentral.com/1471-2105/9/122.
2. HOGEWEG, P, & HESPER, B. (n.d.). *Energy directed folding of RNA sequences*. http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=320984.
3. KITAGAWA, JUNJI, FUTAMURA, YASUHIRO, & YAMAMOTO, KENJI. (n.d.). *Analysis of the conformational energy landscape of human snRNA with a metric based on tree*

representation of RNA structures. Oxford University Press. http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=152804.
4. SCHIRMER, S. (2012). *Comparing forests*. Bielefeld, Universitätsbibliothek Bielefeld, Hochschulschriften. http://nbn-resolving.de/urn:nbn:de:hbz:361-24742380.

1.