

Zadanie 4

Wojciech Miśta, 236453

Działanie programu:

ophilaspAEA2013cy	ATTTATCCACTGATACCCTTTATTTACTGGATTAACATTAAATAATAAATGATTAAAAA	1 - 59
ophilahuancavilca	A_T_T_CACTGGTATCCTTTATTTACTGGATTAACACTTAATAATAAATGATTAAAAA	1 - 59
ophilahuaylasiacyt	ATTCATTTCATTGATACCCTTTATTCACCTGGATTAACCTTTAAATAATAAATTATTAAAAA	1 - 59
ophilaincacytochr	TTTTATTCATTGATACCCTTTATTCACCTGGTCTTACACTAAACAATAAATGATTAAAAA * * * * * ***** * * * * *	1 - 59
ophilaspAEA2013cy	GTCAATTTATCATTATATTTATTTGGAGTAAATTTAACATTTTCCCTCAACATTTTTTA	60 - 118
ophilahuancavilca	GTCAATTTATTTATTTATTTATTTGGAGTAAATTTAACATTTTCCCTCAACATTTTCTA	60 - 118
ophilahuaylasiacyt	GTCAATTTATTTATTTATTTATTTGGTGTAAATTTAACATTTTCCCTCAACATTTTCTT	60 - 118
ophilaincacytochr	GTCAATTTATTTATTTATTTATTTGGAGTAAATTTAACATTTTCCCTCAACATTTTTTA ***** ***** * * ***** ***** *	60 - 118
ophilaspAEA2013cy	GGATTAGCAGGTATACCCGCGCTTATCTGATTATCCTGATGCTTATACAACCTGAAA	119 - 177
ophilahuancavilca	GGGTTAGCGGGTATGCCTCGTCGATATTTCTGATTACCTGATGCTTACACTACATGAAA	119 - 177
ophilahuaylasiacyt	GGATTAGCAGGTATACCTCGACGATATTCAGATTACCCAGATGCTTATACAACATGAAA	119 - 177
ophilaincacytochr	GGGTTAGCTGGGATACCTCGTCGATATTCAGACTACCCGGATGCTTACACTACCTGAAA *	119 - 177
ophilaspAEA2013cy	TGTAATTTCAACAATCGGTTTCAATTTCTTTACTTGGAAATTTATCTTTTTTTTTTA	178 - 236
ophilahuancavilca	TGTAATTTCAACTATTGGGTCTTCAATTTCTTTACTAGGAATTTATCTTTTTTTTATA	178 - 236

Dopasowanie dla gatunków z rodzaju 'Drosophila'.

```
>DrosophilaspAEA2013cy
ATTATCCACTGATACCCTTTATTTACTGGATTAACATTAAATAATAAATGATTAAAAAGTCAATTTATC
ATTATATTTATTTGGAGTAAATTTAACATTTTCCCTCAACATTTTCTAGGATTAGCAGGTATACCCCGCC
GTTATTTCTGATTATCCTGATGCTTATACAACCTGAAATGTAATTTCAACAATCGGTTTCATCAATTTCTTT
ACTTGGAAATTTATCTTTTTTTTATCATTGAGAAAGATTAGTATCCCAACGTCAGTAATTTACCCA
ATTCAATTAATTCATCAATTGAATGATACCAAATACCCCTCCTGCTGAACATAGCTATTCTGAATTAC
CTTTATTAACATAATTAA

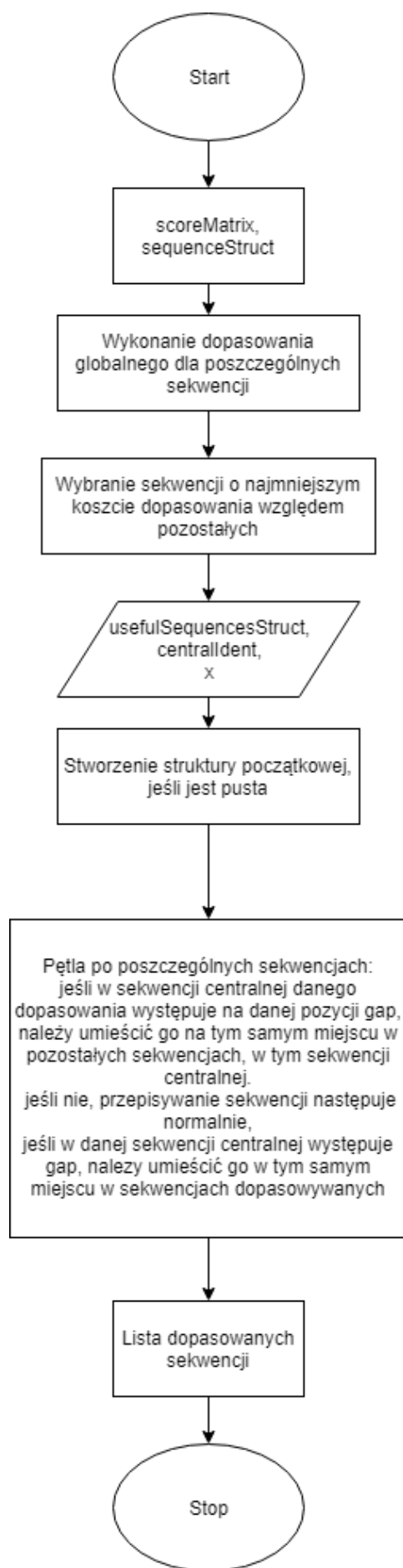
>Drosophilahuancavilca
A_T_T_CACTGGTATCCTTTATTTACTGGATTAACACTTAATAATAAATGATTAAAAAGTCAATTTATT
ATTATATTTATTTGGAGTAAATTTAACATTTTCCCTCAACATTTTCTAGGGTTAGCGGGTATGCCTCGTC
GATATTTCTGATTACCTGATGCTTACACTACATGAAATGTAATTTCAACTATTGGGTCTTCAATTTCTTT
ACTAGGAATTTATCTTTTTTTTATATTATCTGAGAAAGCTTAGTATCCCAACGTCAGTAATTTATCCA
ATTCAATTAATTCATCAATTGAATGGTATCAAATACCCCTCCTGCTGAACATAGTTATTCTGAATTAC
CTCTATTAACATAATT__

>Drosophilahuaylasiacyt
ATTCATTTCATTGATACCCTTTATTCACCTGGATTAACCTTTAAATAATAAATTTAAAAAGTCAATTTATT
ATTATATTTATTTGGTGTAAATTTAACATTTTCCCTCAACATTTTCTGGATTAGCAGGTATACCTCGAC
GATATTCAGATTACCCAGATGCTTATACAACATGAAATGTAATTTCAACAATCGGTTTCATCAATTTTCATT
ATTAGGAATTTATTTTTTTCTATATCATTGAGAAAGTTAGTATCTCAACGTCAGTAATTTATCCA
ATTCAATTAATTCATCTATTGAATGATACCAAATACCTCCTGCGGAACAGATATTCTGAATTAC
CATTATTAACATAATTAA

>Drosophilaincacytochr
TTTTATTCATTGATACCCTTTATTCACCTGGTCTTACACTAAACAATAAATGATTAAAAAGTCAATTTATT
ATTATATTCATTGGAGTAAATTTAACATTTTCCCTCAACATTTTCTAGGTTAGCTGGGATACCTCGTC
GATATTCAGACTACCCGGATGCTTACACTACCTGAAATGTAATTTCAACTATTGGTCTTCAATTTCACT
GTTAGGAATTTATTTTTCTTTATATTATTGAGAAAGTTAGTATCTCAACGTCAGTAATTTATCCA
ATTCAATTAATTCCTCTATCGAATGATACCAAATACCTCCTGCGGAACAGATATTCTGAATTAC
CTTTATTGACTAATTAA
```

Plik .fasta

Liczne konserwacje kolumn w dopasowaniu sekwencji DNA mitochondrialnego gatunków z rodzaju 'Drosophila' ukazuje, zgodnie z założeniem, bliskie powiązanie ewolucyjne, co potwierdza poprawność działania algorytmu.



multipleSequenceAlignm.m

```
function [finalSequenceStruct] = multipleSequenceAlignm(usefulSequencesStruct,x,centralIdent)
%MULTIPLESEQUENCEALIGNMENT Summary of this function goes here
% Detailed explanation goes here
finalSequenceStruct = struct;

%je?li dopasowywany powoduje gapa w centralnym -> dodaj gap do wszystkich
%poza dopasowywanym
%je?li mismatch to nic
%je?li centralny pierwszy ma gap, dodaj gap do dopasowywanego

%loop przez ilosc dopasowan jakie b?d? robione
for i = 1:x
    backup = usefulSequencesStruct;
    %jesli nie ma nic w finalnej strukturze
    if (isempty(fieldnames(finalSequenceStruct)))
        finalSequenceStruct.(centralIdent) = usefulSequencesStruct.("s" + i).(centralIdent);
        usefulSequencesStruct.("s" + i) = rmfield(usefulSequencesStruct.("s" + i),centralIdent);
        lastField = char(fieldnames(usefulSequencesStruct.("s" + i)));
        finalSequenceStruct.(lastField) = usefulSequencesStruct.("s" + i).(lastField);
    else
        %je?li struktury do dopasowania istnieja
        if (isfield(usefulSequencesStruct,char("s"+i)))
            %Centralna sekwencja z danego dopasowania
            usefulCentral = usefulSequencesStruct.("s" + i).(centralIdent);

            %usun z dopasowan zeby uzyskac pozosa??
            usefulSequencesStruct.("s" + i) = rmfield(usefulSequencesStruct.("s" + i),centralIdent);

            %Pozosta?a nazwa field sekwencji z danego dopasowania
            lastField = char(fieldnames(usefulSequencesStruct.("s" + i))); %dopasowywana

            %pozosta?a sekwencja
            lastSequence = char(usefulSequencesStruct.("s" + i).(lastField));

            %Field names w finalnej strukturze
            existingSequencesFields = fieldnames(finalSequenceStruct);

            c = 1;
            currentCentral = char(finalSequenceStruct.(char(existingSequencesFields(1))));

            while numel(char(finalSequenceStruct.(char(existingSequencesFields(1))))) ~= numel(lastSequence)
                %for c = 1: numel(lastSequence)
                currentCentral = char(finalSequenceStruct.(char(existingSequencesFields(1))));
                %patrzmy czy jest gap w starym dopasowaniu
                if (numel(usefulCentral) >= c)
                    if (usefulCentral(c) == '_')
                        %patrzmy czy jest gap w NOWYM dopasowaniu
                        if (currentCentral(c) == '_')
                            %nie rób nic, po prostu dodaj
                        else
                            %nadpisz nowe dopasowanie + wszystkie
                            for n = 1: numel(existingSequencesFields)
                                currentSeq = char(finalSequenceStruct.(char(existingSequencesFields(n))));
                                currentField = char(existingSequencesFields(n));

                                if (c ~= 1)
                                    currentSeq = char(strcat(currentSeq(1:c-1) + "_" + currentSeq(c:end)))
                                elseif (c == numel(currentSeq))
                                    currentSeq = char(strcat(currentSeq + "_"));
                                elseif (c == 1)
                                    currentSeq = char(strcat("_" + currentSeq));
                                end

                                if (c == 1)
                                    currentSeq = char(strcat(currentSeq(1:c-1) + "_" + currentSeq(c:end)))
                                end
                            end
                        end
                    end
                end
            end
        end
    end
end
```

```

        elseif(currentCentral(c) == ' ')
            if(c ~= 1)
                lastSequence = char(strcat(lastSequence(1:c-1) + " " + lastSequence(c:end)));
            elseif(c == numel(lastSequence))
                lastSequence = char(strcat(lastSequence + " "));
            elseif(c == 1)
                lastSequence = char(strcat(" " + lastSequence));
            end
            %dopisz gap do nowej sekwencji
        else
            if(numel(lastSequence) < numel(currentCentral))
                lastSequence = char(strcat(lastSequence + " "));
            end
            %dodac warunek
        end
    end
    c = c + 1;
end
finalSequenceStruct.(lastField) = lastSequence;
end
end
usefulSequencesStruct = backup;
end
end

```

Złożoność obliczeniowa czasowa: $O(kn^2 + k^2l)$

k – sekwencje,

l – górna granica długości sekwencji,

n – czas dopasowania