



Πολυτεχνική Σχολή

Τμήμα Μηχανικών Η/Υ & Πληροφορικής

ΕΙΣΑΓΩΓΗ ΣΤΗ ΒΙΟΠΛΗΡΟΦΟΡΙΚΗ

Βέργος Γεώργιος

A.M. 1072604

Πάτρα, 2024

ΠΕΡΙΕΧΟΜΕΝΑ

Ερώτημα 1	3
Ερώτημα 2	3
Ερώτημα 3	3
Ερώτημα 4	3
Ερώτημα 5	3
Ερώτημα 6	3
Ερώτημα 7	4

Ερώτημα 1

I.

Επισκέπτομαι τον ιστότοπο της Rosalind:



Locations

Rosalind is a platform for learning bioinformatics and programming through problem solving. [Take a tour](#) to get the hang of how Rosalind works.



Python Village



Bioinformatics
Stronghold



Bioinformatics
Armory

If you are completely new to programming, try these initial problems to learn a few basics about the Python programming language. You'll get familiar with the operations needed to start solving bioinformatics challenges in the Stronghold.

Discover the algorithms underlying a variety of bioinformatics topics: computational mass spectrometry, alignment, dynamic programming, genome assembly, genome rearrangements, phylogeny, probability, string algorithms and others.

Ready-to-use software tools abound for bioinformatics analysis. Whereas in the Stronghold you implement algorithms on your own, in the Armory you solve similar problems by using existing tools.

Bioinformatics
Textbook Track



Algorithmic
Heights

A collection of exercises to accompany Bioinformatics Algorithms: An Active-Learning Approach by Philip Compeau & Pavel Pevzner. A full version of this text is hosted on stepic.org

A collection of exercises in introductory algorithms to accompany "Algorithms", the popular textbook by Dasgupta, Papadimitriou, and Vazirani.

Και επιλέγω το 'Bioinformatics Armory' (Έτοιμα εργαλεία τα οποία επιλύουν προβλήματα βιοπληροφορικής χωρίς να απαιτείται να γράψω κώδικα) :

Rosalind Problems Statistics Glossary search [f](#) [t](#) bioeid Log out

Problems

Rosalind is a platform for learning bioinformatics and programming through problem solving. [Take a tour](#) to get the hang of how Rosalind works.

Last win: [rayan.mulla](#) vs. "Variables and Some Arithmetic", 10 minutes ago

Problems: 284 (total), users: 115783

ID	Title	Solved By	Correct Ratio	Questions	Solutions	Explanation
INI	Introduction to the Bioinformatics Armory	7611		2 years	5 months	4 years
GBK	GenBank Introduction	3155				
MEME	New Motif Discovery	1001				
FRMT	Data Formats	2838				
NEED	Pairwise Global Alignment	1370				
TFSQ	FASTQ format introduction	1891				
PHRE	Read Quality Distribution	1313				
PTRA	Protein Translation	1277				
FILT	Read Filtration by Quality	1022				
RVCO	Complementing a Strand of DNA	1094				
SUBO	Suboptimal Local Alignment	549				
BPHR	Base Quality Distribution	759				
CLUS	Global Multiple Alignment	504				
ORFR	Finding Genes with ORFs	741				
BFIL	Base Filtration by Quality	642				

Found a typo? [Take a tour](#)

Κατ' επιλέγω ένα από τα προβλήματα βιοπληροφορικής της παραπάνω λίστας, έστω το πρώτο πρόβλημα με τίτλο: 'Introduction to the Bioinformatics Armory'. Παρακάτω φαίνεται η σελίδα του προβλήματος.

Bioinformatics_first_project_2023-2024

Let's Be Practical click to collapse

If you are an accomplished coder, then you can write a separate program for every new task you encounter. In practice, these programs only need to be written once and posted to the web, where those of us who are not great coders can use them quickly and efficiently. In the Armory, we will familiarize ourselves with a sampling of some of the more popular bioinformatics tools taken from "out of the box" software.

To be equitable, we will focus mainly on free, internet-based software and on programs that are compatible with multiple operating systems. The "Problem" section will contain links to this software, with short descriptions about how to use it.

Problem

This initial problem is aimed at familiarizing you with Rosalind's task-solving pipeline. To solve it, you merely have to take a given DNA sequence and find its nucleotide counts; this problem is equivalent to "Counting DNA Nucleotides" in the [Stronghold](#).

Of the many tools for DNA sequence analysis, one of the most popular is the [Sequence Manipulation Suite](#). Commonly known as SMS 2, it comprises a collection of programs for generating, formatting, and analyzing short strands of DNA and [polypeptides](#).

One of the simplest SMS 2 programs, called [DNA stats](#), counts the number of occurrences of each nucleotide in a given strand of DNA. An online interface for [DNA stats](#) can be found [here](#).

Given: A DNA string s of length at most 1000 bp.

Return: Four integers (separated by spaces) representing the respective number of times that the symbols 'A', 'C', 'G', and 'T' occur in s . **Note:** You must provide your answer in the format shown in the sample output below.

Sample Dataset

```
AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAAGGTGTCTGATAGCAGC
```

Sample Output

```
20 12 17 21
```

Programming Shortcut click to collapse

Our default choice for existing functions and modules to analyze biological data is [BioPython](#), a set of freely available tools for computational biology that are written in [Python](#). We will give you tips on how to solve certain problems (like this one) using BioPython functions and methods.

Detailed installation instructions for BioPython are available in [PDF](#) and [HTML](#) formats.

BioPython offers a specific [data structure](#) called [Seq](#) for representing [sequences](#). [Seq](#) represents an extension of the "str" ([string](#)) object type that is built into Python by supporting additional biologically relevant methods like [translate\(\)](#) and [reverse_complement\(\)](#).

In this problem, you can easily use the built-in Python method [.count\(\)](#) for strings. Here's how you could count the occurrences of 'A' found in a [Seq](#) object.

```
>>> from Bio.Seq import Seq  
>>> my_seq = Seq("AGTACACTGGT")  
>>> my_seq.count("A")
```

Όπου παρουσιάζονται πληροφορίες για το πρόβλημα βιοπληροφορικής π.χ εδώ το πρόβλημα είναι η μέτρηση του αριθμού κάθε βάσης σε μία ακολουθία. Παρέχεται ένα δοκιμαστικό σύνολο δεδομένων και από κάτω γράφεται το αναμενόμενο αποτέλεσμα. Ακόμη παρέχεται και μια προγραμματιστική υπόδειξη για το πρόβλημα. Με την ακόλουθη επιλογή:

Time limit You'll have 5 minutes to upload the answer.

[Questions](#) [Solutions](#) [Explanation](#)

Congratulations You solved this problem (attempt #2). Now you may like to try problems "[GenBank Introduction](#)", "[New Motif Discovery](#)".

[Download dataset](#) You may make an unlimited number of attempts without being penalized.

Γίνεται λήψη του συνόλου δεδομένων του προβλήματος και ξεκινάει μια προθεσμία για την επίλυση του προβλήματος και εμφανίζονται οι παρακάτω επιλογές.

[Download current dataset again](#)

Answer submission 04:53

Copy your answer here:

Or just attach a file with the answer:

No file selected.

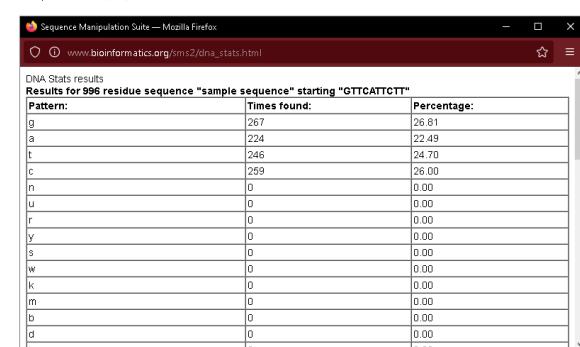
Your code (optional):

No file selected.

Όταν επιλυθεί το συγκεκριμένο πρόβλημα μέσω κώδικα ή των έτοιμων εργαλείων που παρέχει το armory εισάγουμε στο πλαίσιο ελεύθερου κειμένου την έξιδο του προγράμματος που φτιάχαμε/χρησιμοποιήσαμε και το σύστημα εμφανίζει εάν η λύση μας είναι σωστή ή όχι. Ακόμη στην παράγραφο 'Problem' παρέχεται ένας σύνδεσμος σε ένα έτοιμο εργαλείο το οποίο λύνει το πρόβλημα αυτό:

[Format Conversion](#)
[Convert to FASTA](#)
[EMBL Feature Extractor](#)
[EMBL Feature Validator](#)
[Filter DNA](#)
[Filter protein](#)
[General FASTA](#)
[General FASTA Extractor](#)
[GenBank Feature Extractor](#)
[GenBank Feature Validator](#)
[One to Three](#)
[Range DNA](#)
[Range Feature Protein](#)
[Reverse Complement](#)
[Split DNA](#)
[Split FASTA](#)
[Three to One](#)
[Window Extractor DNA](#)
[Window Extractor Protein](#)
[Sequence Analysis](#)
[Codon Plot](#)
[Codon Usage](#)
[Codon Wheel](#)
[DNA Molecular Weight](#)
[DNA Pattern Find](#)
[DNA Stats](#)
[Fuzzy Search DNA](#)
[Fuzzy Search Protein](#)
[Ident and Sim](#)
[Multi-Freq Trans](#)
[Multiple Sequence](#)
[ORF Finder](#)
[Protein Align Codons](#)
[Protein Align DNA](#)
[Protein Align Protein](#)
[PDB Search](#)
[PCR Products](#)
[Phenotype Query](#)
[Protein Isoelectric Point](#)
[Protein Molecular Weight](#)
[Protein Pattern Find](#)
[Protein Stats](#)
[Redaction Toolkit](#)
[Redirection Summary](#)
[Reverse Tandem](#)
[Tandem](#)
[Sequence Figures](#)
[Color Align Conservation](#)
[Color Align Properties](#)
[Group DNA](#)

Sun, 1 Oct 12:00:09 2023
Valid HTML 1.0, Valid CSS



Όπου εισάγοντας το σύνολο δεδομένων του προβλήματος στο εργαλείο αυτό

και πατώντας 'Submit' εμφανίζονται τα στατιστικά στοιχεία της ακολουθίας συμπεριλαμβανομένων και αυτών που απαντεί το πρόβλημα.

Ακόμη, για παράδειγμα για το πρόβλημα ολικής στοίχισης (Pairwise global alignment) δύο ακολουθιών χρησιμοποιώ το εργαλείο **EMBOSS Needle**:

Pairwise Global Alignment solved by 1370

March 22, 2013, 9:10 p.m. by Rosalind Team

Topics: Alignment, Bioinformatics Tools

Comparing Strings Online click to expand

Problem

An online interface to EMBOSS's [Needle](#) tool for aligning DNA and RNA strings can be found [here](#).

Use:

- The [DNAfull](#) scoring matrix; note that DNAfull uses IUPAC notation for ambiguous nucleotides.
- Gap opening penalty of 10.
- Gap extension penalty of 1.

For our purposes, the "pair" output format will work fine; this format shows the two strings aligned at the bottom of the output file beneath some statistics about the alignment.

Given: Two GenBank IDs.

Return: The maximum global alignment score between the DNA strings associated with these IDs.

Sample Dataset

```
JX205496.1 JX469991.1
```

Sample Output

```
257
```

Programming Shortcut click to expand

Global Alignment

Global alignment tools create an end-to-end alignment of the sequences to be aligned.

EMBOSS Needle

EMBOSS Needle creates an optimal global alignment of two sequences using the Needleman-Wunsch algorithm.

[Launch EMBOSS Needle](#)

To εργαλείο φαίνεται στην παρακάτω εικόνα:

Bioinformatics_first_project_2023-2024

Job Dispatcher Help & Privacy Your Jobs Input form Feedback

Welcome to the new Job Dispatcher website. We'd love to hear your feedback about the new webpages! x

EMBOSS Needle reads two input sequences and writes their optimal global sequence alignment to file.

Input sequence ? **Sequence type** ?

Protein DNA

Paste your first sequence here - or use the example sequence

No file selected.

Paste your second sequence here - or use the example sequence

No file selected.

Parameters

OUTPUT FORMAT ?

Submit

Title

Όπου μπορούμε να εισάγουμε τις ακολουθίες στα πλαίσια και να πάρουμε την ολική τους στοίχιση.

Αντίστοιχα δουλεύουμε στα υπόλοιπα εργαλεία των υπόλοιπων προβλημάτων βιοπληροφορικής.

II.

Η βάση βιολογικών δεδομένων **EBI** περιλαμβάνει τα εξής εργαλεία πολλαπλής στοίχισης ακολουθιών:

Explore Sequence Analysis Tools with **Job Dispatcher** EMBL's European Bioinformatics Institute

Job Dispatcher Help & Privacy Job Dispatcher Feedback

Multiple Sequence Alignment

Multiple Sequence Alignment (MSA) is generally the alignment of three or more biological sequences (protein or nucleic acid) of similar length. From the output, homology can be inferred and the evolutionary relationships between the sequences studied. By contrast, Pairwise Sequence Alignment tools are used to identify regions of similarity that may indicate functional, structural and/or evolutionary relationships between two biological sequences.

Clustal Omega

New MSA tool that uses seeded guide trees and HMM profile-profile techniques to generate alignments. Suitable for medium-large alignments.

EMBOSS Cons

EMBOSS Cons creates a consensus sequence from a protein or nucleotide multiple alignment.

Kalign

Very fast MSA tool that concentrates on local regions. Suitable for large alignments.

Bioinformatics_first_project_2023-2024

MAFFT

MSA tool that uses Fast Fourier Transforms. Suitable for medium-large alignments.

[Launch MAFFT](#)

MUSCLE

Accurate MSA tool, especially good with proteins. Suitable for medium alignments.

[Launch MUSCLE](#)

MView

Transform a Sequence Similarity Search result into a Multiple Sequence Alignment or reformat a Multiple Sequence Alignment using the MView program.

[Launch MView](#)

T-Coffee

Consistency-based MSA tool that attempts to mitigate the pitfalls of progressive alignment methods. Suitable for small alignments.

[Launch T-Coffee](#)

WebPRANK

The EBI has a new phylogeny-aware multiple sequence alignment program which makes use of evolutionary information to help place insertions and deletions.

[Launch WebPRANK](#)

Το εργαλείο **T-coffee** χρησιμοποιείται και στο ερώτημα 2 της εργασίας. Είναι ένα σύγχρονο εργαλείο πολλαπλής στοίχισης ακολουθιών το οποίο μετριάζει τα προβλήματα που παρουσιάζουν άλλες σύγχρονες μέθοδοι πολλαπλής στοίχισης ακολουθιών. Συνίσταται για μικρές στοιχίσεις. Αντίθετα το **Kalign** είναι ένα γρήγορο εργαλείο πολλαπλής στοίχισης ακολουθιών το οποίο δουλεύει καλά για μεγάλου μεγέθους στοιχίσεις. Το **MUSCLE** προτείνεται για μεσαίου μεγέθους στοιχίσεις και ιδιαίτερα όταν οι ακολουθίες είναι πρωτεΐνες. Για μεσαίου (ή και μεγαλύτερου) μήκους στοιχίσεις αποδοτικά είναι επίσης τα εργαλεία **MAFFT**, **Clustal Omega**. Το **MUSCLE** πετυχαίνει μεγαλύτερη ακρίβεια και απόδοση (ταχύτητα) από τα εργαλεία **Clustal W** και **T-Coffee**. Τα εργαλεία **Clustal Omega**, **T-Coffee** υποστηρίζουν και στοίχιση **RNA** ακολουθιών. Το εργαλείο **Mview** λαμβάνει το αποτέλεσμα πολλαπλής στοίχισης από κάποιο άλλο εργαλείο και τα οπτικοποιεί με καλόγονυστο τρόπο, δίνοντας περισσότερες πληροφορίες για τη στοίχιση. Δεν είναι πρόγραμμα πολλαπλής στοίχισης αλλά μόνο οπτικοποίησης όπως το **MSA viewer NCBI**.

Η βάση βιολογικών NCBI διαθέτει δύο κύρια εργαλεία πολλαπλής στοίχισης ακολουθιών, με πρώτο το **MSA Viewer**, το οποίο είναι

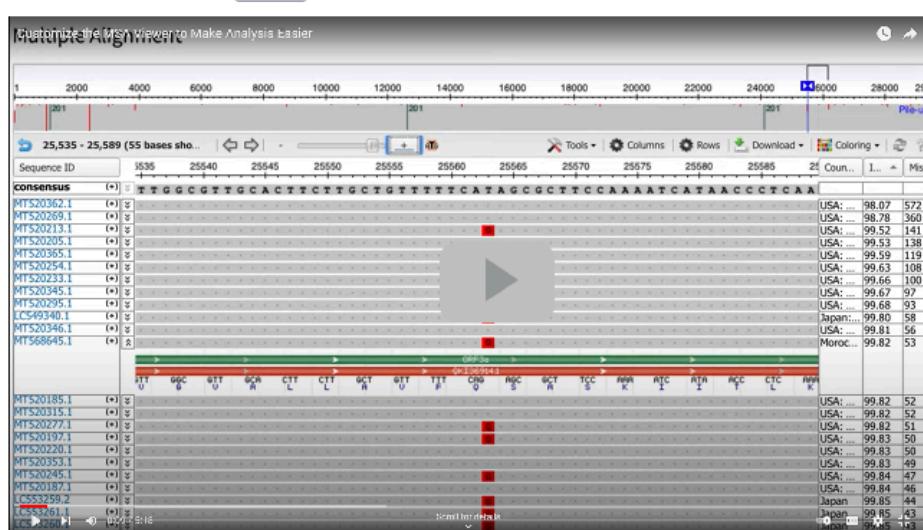
μία γραφική διεπαφή για πολλαπλή στοίχιση πρωτεΐνών αλλά και άλλων βιολογικών ακολουθιών.

NCBI Multiple Sequence Alignment Viewer

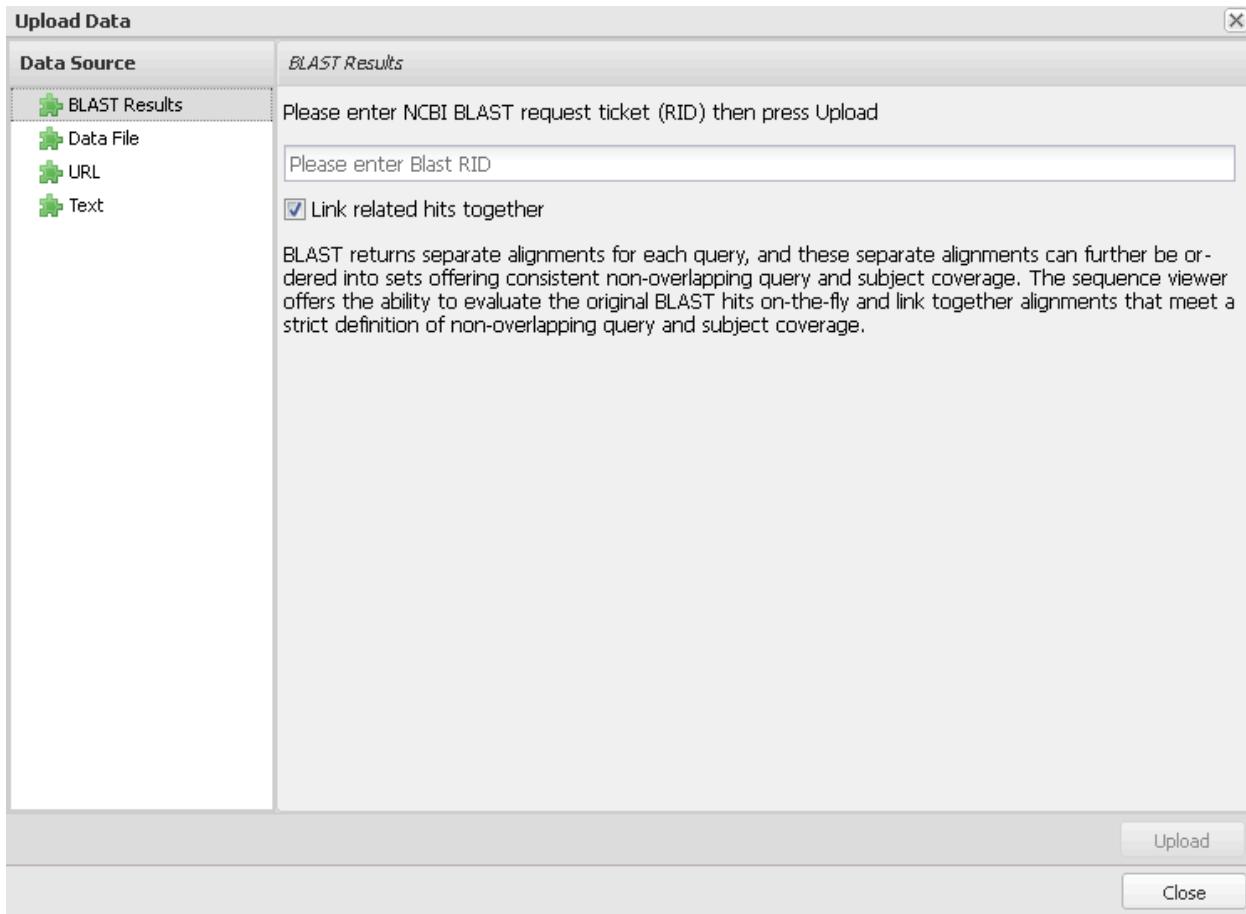
The NCBI Multiple Sequence Alignment Viewer (MSA) is a graphical display for nucleotide and protein sequence alignments.

Review [documentation](#) or watch [a video tutorial](#).

To see your own alignment, [Upload](#) your data

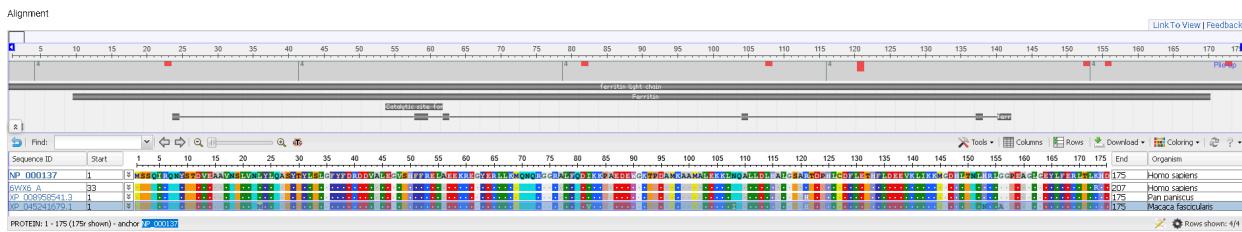


Στην παραπάνω εικόνα φαίνεται η αρχική σελίδα του **MSA Viewer** όπου παρουσιάζονται βιοηθητικές πληροφορίες για το εργαλείο όπως η τεκμηρίωση κλπ. Με το κουμπί 'upload' η εφαρμογή ανακατευθύνει στην παρακάτω εικόνα:



Όπου μπορούμε να επιλέξουμε είτε τη μεταφόρτωση των ακολουθιών από ένα αρχείο fasta, να εισάγουμε τις ίδιες τις ακολουθίες χειροκίνητα ή ακόμα και να τις εξάγουμε από κάποια παλαιότερη αναζήτηση που κάναμε για αυτές στην **NCBI**.

Μεταφορτώνοντας ένα asn αρχείο (που περιέχει έστω 3 πρωτεΐνες) το πρόγραμμα εμφανίζει τα αποτελέσματα της στοίχισης:



Το δεύτερο εργαλείο πολλαπλής στοίχισης της NCBI είναι το **COBALT** το οποίο φαίνεται παρακάτω.

COBALT Constraint-based Multiple Alignment Tool

Enter Query Sequences
Enter at least 2 protein accessions, gis, or FASTA sequences COBALT computes a multiple protein sequence alignment using conserved domain and local sequence similarity information.

Or, upload FASTA file No file selected.

Job Title:

Align Show results in a new window

Advanced parameters

Alignment Parameters

Gap Penalties: Opening -11, Extension -1

End-Gap Penalties: Opening -5, Extension -1

Constraint Parameters

RPS Blast: Use RPS BLAST to guide alignment

Constraint E-value: 0.005

Conserved columns: Find Conserved Columns and Recompute Alignment

Query Clustering Parameters

Query Clustering Parameters: Use query clusters

Word Size: 4

Max Cluster distance: 0.8

Alphabet: SE-815

Align Show results in a new window

Το εργαλείο όπως βλέπουμε έχει ως κοινό χαρακτηριστικό με το **MSA viewer** τους τρόπους εισαγωγής των ακολουθιών στο εργαλείο όπως μεταφόρτωση αρχείου fasta, χειροκίνητη εισαγωγή των ακολουθιών, υποστήριξη γραφικού περιβάλλοντος κλπ. Σε αντίθεση με τον **MSA Viewer** διαθέτει δυνατότητα εισαγωγής επιπλέον παραμέτρων όπως για παράδειγμα κόστη στοίχισης με κενό και άλλες που φαίνονται παραπάνω. Στην κάτω εικόνα φαίνεται για τις ίδιες ακολουθίες το αποτέλεσμα της στοίχισης τους:

COBALT Constraint-based Multiple Alignment Tool

Phylogenetic Tree Home Recent Results Help

- Cobalt RID DVAZ79G8212 (3 seqs)

Graphical Overview

PROTEIN: 1 - 227 (227 shown)

Descriptions Select All

Accession	Description	Links
IClQuery_10001	6W6_A Chain A, Ferritin light chain [Homo sapiens]	
IClQuery_10002	XP_008958541.3 ferritin light chain [Pan paniscus]	
IClQuery_10003	XP_045241679.1 ferritin light chain [Macaca fascicularis]	

Alignments Select All Mouse over the sequence identifier for sequence title

View Format: Compact Conservation Setting: 2 Bits

Query_10001 1	[32]HSQIQRNYSTVNEAHNSLVNLYQASYYTLSLGFYFDRDVVALEGVSFFRELAEKREGYERLLKHQNQRRR	108
Query_10002 1	HSQIQRNYSTVNEAHNSLVNLYQASYYTLSLGFYFDRDVVALEGVSFFRELAEKREGYERLLKHQNQRRR	76
Query_10003 1	HSQIQRNYSTVNEAHNSLVNLYQASYYTLSLGFYFDRDVVALEGVSFFRELAEKREGYERLLKHQNQRRR	76

Το **MSA viewer** είναι αρκετά πιο διαδραστικό και σύνθετο συγκριτικά με το **COBALT** και παρέχει περισσότερες πληροφορίες για την στοιχιση γεγονός που φαίνεται στις παραπάνω εικόνες. Στην πραγματικότητα το **MSA viewer** δεν στοιχίζει τις ακολουθίες αλλά απλώς απεικονίζει στοιχίσεις ακολουθιών που έχουν ήδη πραγματοποιηθεί. Αυτό το καταλαβαίνει κανείς στην μεταφόρτωση του αρχείου fasta των ακολουθιών. Προσπάθησα να μεταφορτώσω το αρχείο fasta των ακολουθιών που επέλεξα στο ερώτημα 2 αφού τις κατέβασα ως fasta format (complete sequence) στο **MSA viewer** και το αρχείο απορρίφηκε. Αυτό έγινε γιατί οι ακολουθίες δεν ήταν στοιχισμένες και έπρεπε πρώτα να τις κάνω λήψη σε κάποια στοιχισμένη μορφή όπως **asn** ή **aligned sequence fasta**.

Ερώτημα 2

Στην ιστοσελίδα πραγματοποιώ μία **BLASTP** (protein-to-protein) αναζήτηση από τη βάση βιολογικών δεδομένων **NCBI**, χρησιμοποιώντας για κωδικό αναζήτησης την ελαφριά αλυσίδα της φερριτίνης (**NP_000137**). Η αναζήτηση αυτή φαίνεται στην παρακάτω εικόνα:

Standard Protein BLAST

Enter Query Sequence

Enter accession number(s), gi(s), or FASTA sequence(s)

From
To

Or, upload file No file selected.

Job Title

Align two or more sequences

Choose Search Set

Databases Standard databases (nr etc.) Experimental databases
 Compare Select to compare standard and experimental database

Standard

Database
 Organism exclude
 Exclude Models (XM/XP) Non-redundant RefSeq proteins (WP) Uncultured/environmental sample sequences

Program Selection

Algorithm Quick BLASTP (Accelerated protein-protein BLAST)
 blastp (protein-protein BLAST)
 PSI-BLAST (Position-Specific Iterated BLAST)
 PHI-BLAST (Pattern Hit Initiated BLAST)
 DELTA-BLAST (Domain Enhanced Lookup Time Accelerated BLAST)
 Choose a BLAST algorithm

BLAST Show results in a new window

Πατώνοντας την λέξη **'BLAST'** στην παραπάνω εικόνα, πραγματοποιώ την ζητούμενη αναζήτηση και εμφανίζονται τα ακόλουθα αποτελέσματα:

National Library of Medicine National Center for Biotechnology Information

BLAST® » blastp suite » results for RID-9V5881G3013

Job Title	NP_000137:ferritin light chain [Homo sapiens]
RID	9V5881G3013 <small>Search expires on 07-23 07:08 am</small> <input type="button" value="Download All"/>
Program	BLASTP <input type="button" value="Citation"/>
Database	nr <input type="button" value="See details"/>
Query ID	NP_000137.2
Description	ferritin light chain [Homo sapiens]
Molecule type	amino acid
Query Length	175
Other reports	Distance tree of results Multiple alignment MSA viewer <input type="button" value="?"/>

Filter Results

Organism only top 20 will appear exclude
 Type common name, binomial, taxid or group name

Percent Identity	E value	Query Coverage
<input type="text"/> to <input type="text"/>	<input type="text"/> to <input type="text"/>	<input type="text"/> to <input type="text"/>
<input type="button" value="Filter"/>	<input type="button" value="Reset"/>	

Επειδή οι επιστρεφόμενες ακολουθίες είναι 100 παρουσιάζω στην παρακάτω εικόνα μόνο μερικές από αυτές:

Sequences producing significant alignments		Download		Select columns		Show		100	?				
	Description	GenPept	Graphics	Distance tree of results	Multiple alignment	MSA Viewer	Max Score	Total Score	Query Cover	E value	Per. Ident	Acc. Len	Accession
<input checked="" type="checkbox"/>	Chain A_Ferritin light chain [Homo sapiens]	Homo sapiens	363	363	100%	2e-125	99.43%	227	GWXG_A				
<input checked="" type="checkbox"/>	ferritin light chain [Homo sapiens]	Homo sapiens	360	360	100%	3e-125	100.00%	175	NP_000137.2				
<input checked="" type="checkbox"/>	Homo sapiens ferritin_light polypeptide [synthetic construct]	synthetic construct	360	360	100%	4e-125	100.00%	176	AAP36762.1				
<input checked="" type="checkbox"/>	ferritin-like domain-containing protein [Pseudomonas aeruginosa]	Pseudomonas aeruginosa	361	361	100%	8e-125	100.00%	237	WP_217683847.1				
<input checked="" type="checkbox"/>	hypothetical protein [Homo sapiens]	Homo sapiens	361	361	100%	1e-124	100.00%	241	CAE11873.1				
<input checked="" type="checkbox"/>	ferritin light subunit [Homo sapiens]	Homo sapiens	358	358	100%	1e-124	99.43%	175	AAA35831.1				
<input checked="" type="checkbox"/>	FTL [Homo sapiens]	Homo sapiens	358	358	100%	1e-124	99.43%	175	CAG32996.1				
<input checked="" type="checkbox"/>	FTL [synthetic construct]	synthetic construct	358	358	100%	1e-124	99.43%	175	AKI70338.1				
<input checked="" type="checkbox"/>	FTL [synthetic construct]	synthetic construct	358	358	100%	2e-124	99.43%	175	AIC54405.1				
<input checked="" type="checkbox"/>	FTL [synthetic construct]	synthetic construct	358	358	100%	2e-124	99.43%	175	AKI70336.1				
<input checked="" type="checkbox"/>	Chain A_Ferritin light chain [Homo sapiens]	Homo sapiens	358	358	99%	2e-124	100.00%	174	2FG4_A				
<input checked="" type="checkbox"/>	Ferritin_light polypeptide [Homo sapiens]	Homo sapiens	358	358	100%	2e-124	99.43%	175	AAH16715.1				
<input checked="" type="checkbox"/>	ferritin light chain [Pan paniscus]	Pan paniscus	357	357	100%	3e-124	99.43%	175	XP_008958541.3				
<input checked="" type="checkbox"/>	Ferritin_light polypeptide [Homo sapiens]	Homo sapiens	357	357	100%	4e-124	99.43%	175	AAH13928.1				

Επιλέγω 3 ακολουθίες έστω τις: <> και τις κάνω λήψη ως αρχείο fasta (complete sequence).

I.

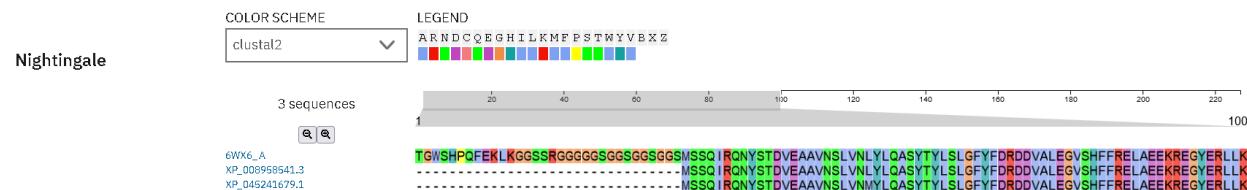
Έπειτα με τη χρήση του εργαλείου **T-Coffee** του **EBI** στοιχίζω τις 3 πρωτεΐνες πατώντας 'Submit':

T-Coffee is a multiple sequence alignment program. The main characteristic of T-Coffee is that it will allow you to combine results obtained with several alignment methods. Important note: This tool can align up to 500 sequences or a maximum file size of 1 MB.

Input sequence ①	Sequence type		
	<input checked="" type="radio"/> Protein <input type="radio"/> DNA <input type="radio"/> RNA		
Paste your sequence here - or use the example sequence			
<input type="button" value="Browse..."/> akolouthies.fasta			
<input type="button" value="Use the example"/> <input type="button" value="Clear sequence"/> More example inputs			
Parameters	OUTPUT FORMAT ②	MATRIX ③	ORDER ④
	<input type="button" value="HTML"/> <input type="button" value="None"/>	<input type="button" value="aligned"/>	
Less options ▾			
Submit	Title <input type="text" value=""/>		

Τα αποτελέσματα της στοιχίσης φαίνονται παρακάτω:

Bioinformatics_first_project_2023-2024



Tool output

Download

T-COFFEE, Version_13.46.0.919e8c6b (2023-07-07 22:06:42 - Revision 29996c5 - Build 980)

Cedric Notredame

CPU TIME: 0 sec.

SCORE=1000

*

BAD AVG GOOD

*

6WX6_A : 100
XP_008958541.3 : 100
XP_045241679.1 : 100
cons : 100

6WX6_A TGWSHPQFEKLKGSSRGGGGSGGSGGSMSSQ|RQNYSTDV
XP_008958541.3 -----MSSQ|RQNYSTDV
XP_045241679.1 -----MSSQ|RQNYSTDV

cons *****

6WX6_A EAAVNSLVNLQASYTYSLSLGFYFDRDDVALEGVS~~HFF~~RELAEE
XP_008958541.3 EAAVNSLVNLQASYTYSLSLGFYFDRDDVALEGVS~~HFF~~RELAEE
XP_045241679.1 EAAVNSLVNMYLQASYTYSLSLGFYFDRDDVALEGVS~~HFF~~RELAEE

cons *****

6WX6_A KREGYERLLKMQNQRGGRALFQDIIKKPAEDEWGKTPDAMKAAMAL
XP_008958541.3 KREGYERLLKMQNQRGGRALFQDIIKKPAEDEWGKTPDAMKAAMAL
XP_045241679.1 KREGYERLLKMQNQRGGRALFQDVKKPAEDEWGKTPDAMKAAMAL

cons *****

6WX6_A EKKLNQALLDLHALGSARTDPHLCDFLETHFLDEEVKLIKKMGDH
XP_008958541.3 EKKLNQALLDLHALGSAHTDPHLCDFLETHFLDEEVKLIKKMGDH
XP_045241679.1 EKKLIQALLDLHALGSAHTDPHLCDFLETHFLDEEVKLIKKMGDH

cons **** *****

6WX6_A LTNLHRLGGPEAGLGEYLFERLTLRHDGGSGGGGGGGGGASG
XP_008958541.3 LTNLHRLGGPEAGLGEYLFERLTLKHD-----
XP_045241679.1 LTNLNRLAGPEAGLGEYLFERLTLKHD-----

cons ***:*,*****:***

6WX6_A GS

XP_008958541.3 --

XP_045241679.1 --

cons

Bioinformatics first project 2023-2024

Με βαθμολογία 1000.

II.

Χρησιμοποιώντας το εργαλείο swiss-modeller βλέπω τη δομή των πρωτεΐνών.

Untitled Project		Created: yesterday at 21:36	
Summary	Templates 50	Models 1	
Project Summary			
Target 1	TGSMSRPDKLGSSSRGDGSSSGS5GSGSNSSQIRQNYSTDVEAVNSLVNLYLQASYYTLSLIGFYEDRODVALEGVSHFFELAEEKREGYERLLKQNQRGGRALFQDIKKPAEDEWSKTFDANKAAHALEK	195	
Target 1	EYLFERULTRHD985504500596000A3003	227	
Template Results <small>•</small>			
A total of 1661 templates were found to match the target sequence. This list was filtered by a heuristic down to 50. The top templates are:			
Template	Sequence Identity	Bisubunit Oligo State	Description
8b70_1	53.89	homo-24-mer	Ferritin heavy chain, N-terminally processed X-ray structure of Auranofin-human H-chain ferritin
8lkh2_1	85.47	homo-24-mer	Ferritin light chain Amyloid-beta precursor protein (Fragment) Crystal structure of horse spleen L-ferritin fused with amyloid beta peptide (1-42).
1fb3_1	81.50	homo-24-mer	FERRITIN LIGHT CHAIN 1 Structure of recombinant mouse L chain ferritin at 1.2 Å resolution
1x21_1	86.78	homo-24-mer	Ferritin light chain Complex of halothane with apoferritin
6zdd_1	81.14	homo-24-mer	Ferritin L-FerritinMSA
Show full template details			
Model Results <small>•</small>			

Για τη δεύτερη:

Bioinformatics_first_project_2023-2024

Untitled Project Created: today at 21:40

Summary Templates 41 Models 1 Project Data

Model Results 1 Order by: GMQE

Model 01

Structure Assessment Compare Download files Display files

Oligo-State: Homo-24-mer (matching prediction) GMQE: 0.91 OMEANDisCo Global: 0.88 ± 0.05

Ligands: 24 x CD¹²

OMEANDisCo Local OMEAN Z-Scores

Template: 1B3.1 X FERRITIN LIGHT CHAIN 1 Structure of recombinant mouse L chain ferritin at 1.2 Å resolution Seq Identity: 81.03% Coverage:

Model-Template Alignment

Cartoon ▲ □ ▶ ▲ ▲ ▲

Untitled Project Created: yesterday at 21:40

Summary Templates 41 Models 1 Project Data

Project Summary

Target sequence: MSLQIIRNYSTIDVEAVVNLVLVNLQLQASYYTLSSLGPFDRDDVALAEVSHFFRELAEEXXREGYERLLVHNQNQNSGRALFQDIIKKPAEDEDWGTTPDANKKANALEKKLNQALLDLHALGSAHTDPHLCDFLETHFLDEEVKLIXXNGDHLTNLHRLGSPEAGLGEYLFERLTLEND 175

Template Results 0

A total of 1271 templates were found to match the target sequence. This list was filtered by a heuristic down to 41. The top templates are:

Template	Sequence identity	Blouin Oligo State	Description
1B3.1	81.03	homo-24-mer	FERRITIN LIGHT CHAIN 1 Structure of recombinant mouse L chain ferritin at 1.2 Å resolution
2fk.1	99.42	homo-24-mer	ferritin light chain Structure of Human Ferril L. Chain
1B3.1	81.03	homo-24-mer	FERRITIN LIGHT CHAIN 1 Structure of recombinant mouse L chain ferritin at 1.2 Å resolution
5igb.1	99.43	homo-24-mer	Ferritin light chain Human L-type ferritin iron loaded for 60 minutes
2v2n.1	87.36	homo-24-mer	FERRITIN LIGHT CHAIN Mutant R59M recombinant horse spleen apoferritin cocrystallized with haemin in acidic conditions

Show full template details

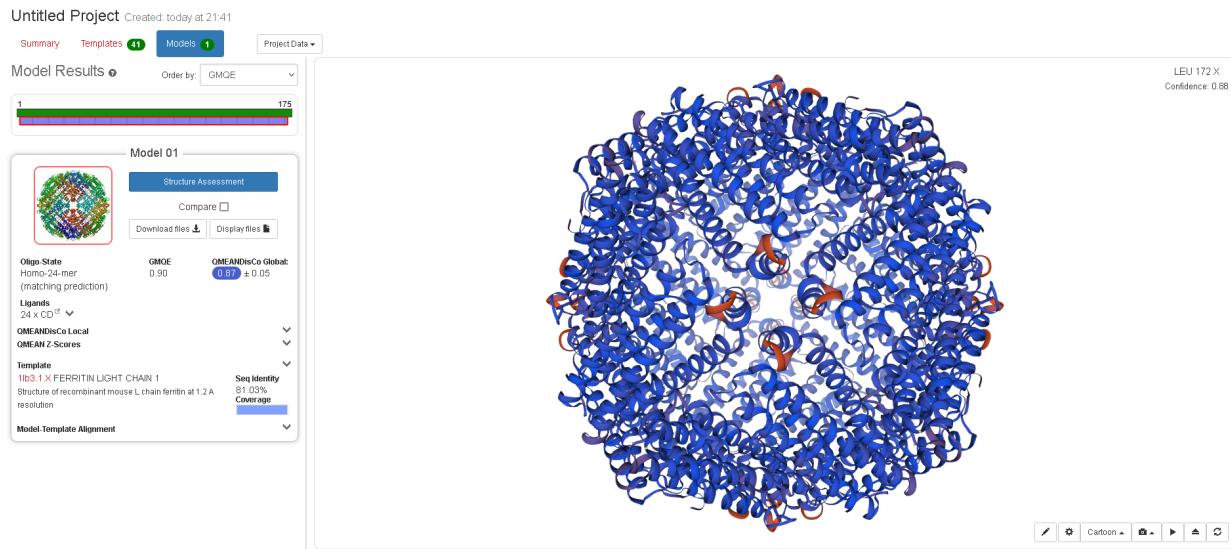
Model Results 0

ID	Template	GMQE	OMEANDisCo Global	Oligo State	Ligands
01	1B3.1 X	0.91	0.88 ± 0.05	homo-24-mer	24 x CD ¹²

Show full model results

Τι α την τρίτη:

Bioinformatics_first_project_2023-2024



Untitled Project Created: yesterday at 21:41

Summary Templates 41 Models 1 Project Data ▾

Project Summary

Target 1: NSQIIRNNTIDVEAAVNLVNNYLQASYITYLQLGIFYFDDOVALEGVSHFFRELACEEKRESYERLLKQNQQRGGRALFQDVVKFAEDEWSKTPDANYKANALEKKLIQALLDLHALGSARTPHLCDFLETHFLDEEVKLIXXNGDHLTNLNLAQPEAGLGEYLFERLTLEMU 175

Template Results

A total of 1307 templates were found to match the target sequence. This list was filtered by a heuristic down to 41. The top templates are:

Template	Sequence Identity	Biounit Oligo State	Description
1lb3.1	81.03	homo-24-mer	FERRITIN LIGHT CHAIN 1 Structure of recombinant mouse L chain ferritin at 1.2 Å resolution
1lb3.1	81.03	homo-24-mer	FERRITIN LIGHT CHAIN 1 Structure of recombinant mouse L chain ferritin at 1.2 Å resolution
2fkh.1	96.53	homo-24-mer	ferritin light chain Structure of Human Ferritin L Chain
6z3d.1	81.03	homo-24-mer	Ferritin L-FerritinM6A
2v2n.1	86.21	homo-24-mer	FERRITIN LIGHT CHAIN Mutant R59M recombinant horse spleen apoferritin cocrystallized with haemin in acidic conditions

Show full template details

Model Results

ID	Template	GMQE	QMAlignCo Global	Oligo State	Ligands
01	1lb3.1 X	0.90	0.87 ± 0.05	homo-24-mer	24 x CD ¹²

Show full model results

Και τέλος κατεβάζω τα .pdb αρχεία:

Download files Display

- PDB format
- ModelCIF format
- DeepView format
- Model Report
- Metadata

Delete Model

Με το εργαλείο Dali συγκρίνω τις δομές των 3 προηγούμενων πρωτεΐνων. Συγκεκριμένα, επιλέγω την εφαρμογή '**All against all structure comparison**' για να συγκρίνω όλες τις πρωτεΐνες μεταξύ τους.

PROTEIN STRUCTURE COMPARISON SERVER

About PDB search PDB25 AF-DB search Pairwise All against all Tutorials References Statistics Download

All against all structure comparison

STEP 1 - Enter your input protein structures

Use the +/- buttons to create input fields. Structures may be specified by concatenating the PDB identifier (4 characters) and a chain identifier (1 character) or, alternatively, you may upload a PDB file. PDB files should be entered before PDB identifiers. The maximum number of input structures is 64. If your input set consists only of structures in the PDB, you can use [this alternative submission form](#).

[+] [-]

STEP 2 - Optional data

You may leave an e-mail address for notification when the job has finished. The job title is used as subject heading in the e-mail.

Job title
E-mail

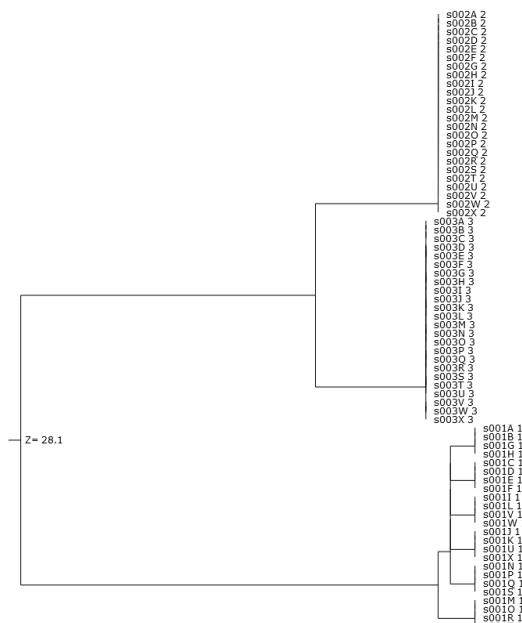
STEP 3 - Submit your job

Submit Clear

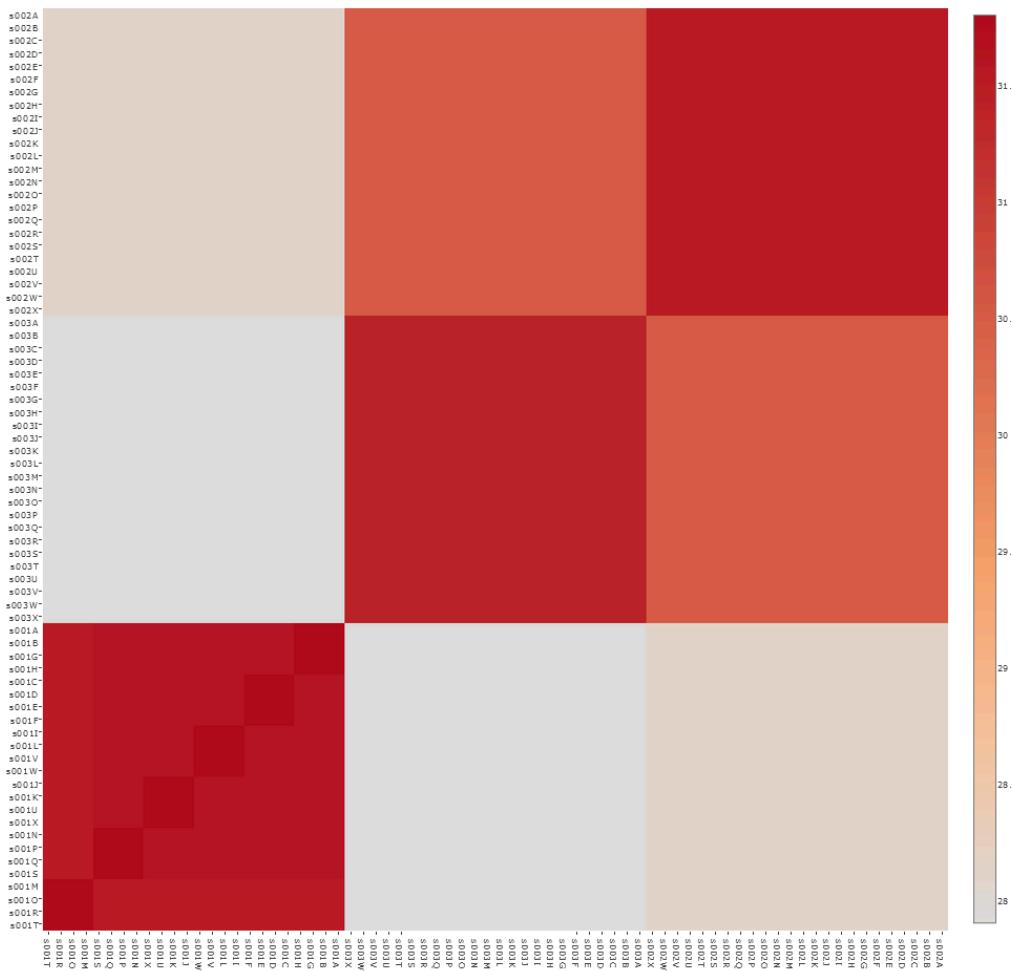
Έπειτα μεταφορτώνω ένα-ένα (πατώντας το '+') τα **pdb** αρχεία που έκανα λήψη στο προηγούμενο ερώτημα και πατάω Submit.
Παρακάτω φαίνεται το τελικό αποτέλεσμα σύγκρισης των δομών των 3 πρωτεΐνων:

Bioinformatics_first_project_2023-2024

Structural similarity dendrogram. Labels are linked to structural summaries. The dendrogram is derived by average linkage clustering of the structural similarity matrix (Dali Z-scores).



Κατ το αντίστοιχο heatmap:



Ερώτημα 3

Ένα δέντρο επιθεμάτων μπορεί να γενικευτεί έτσι ώστε να αναπαριστά ένα σύνολο από συμβολοσειρές τέτοιο ώστε να μπορούν να εισάγονται και να διαγράφονται συμβολοσειρές αποδοτικά από αυτό. Οι υπάρχοντες αλγόριθμοι εισαγωγής και διαγραφής συμβολοσειρών (π.χ. *mcCreight*) ενώ είναι αποδοτικοί, παρουσιάζουν πρόβλημα όταν αφαιρείται κάποια συμβολοσειρά, διότι δεν επαρκεί ο χρόνος ώστε να ενημερωθούν κατάλληλα οι ετικέτες των ακμών του δέντρου που σχετίζονται τη συγκεκριμένη συμβολοσειρά. Γι' αυτό το λόγο, όλες οι συμβολοσειρές που προστίθενται πρέπει να παραμένουν στην κύρια μνήμη. Αυτό έχει δυστυχώς ως αποτέλεσμα να αυξάνονται οι χωρικές απαιτήσεις. Αναλυτικότερα, όταν μία συμβολοσειρά x_j διαγράφεται από ένα σύνολο συμβολοσειρών S , διαγράφονται

μόνο τα φύλλα που σχετίζονται με τα επιθέματα της x_j ενώ δεν μπορούν να εντοπιστούν και να ενημερωθούν έγκαιρα οι ετικέτες των εσωτερικών ακμών που σχετίζονται με υποσυμβολοσειρές της x_j . Έτσι, αυτές οι ακμές και όλες οι αντίστοιχες ακμές των υπόλοιπων συμβολοσειρών που έχουν διαγραφεί από το S πρέπει να παραμείνουν στην κύρια μνήμη γιατί δεν μπορεί να γνωρίζει ο αλγόριθμος αν αυτή η εσωτερική ακμή αναφέρεται σε υποσυμβολοσειρά της x_j . Αυτό δημιουργεί προβλήματα χώρου μιας και η χωρική πολυπλοκότητα δεν μπορεί να φραχθεί από το συνολικό μήκος των συμβολοσειρών στο S.

Αυτό μπορεί να επιλυθεί με μία νέα εκδοχή του δέντρου επιθεμάτων, το **δυναμικό δέντρο επιθεμάτων**, στο οποίο δεν αποθηκεύονται αναφορές σε συμβολοσειρές που έχουν διαγραφεί. Στο δέντρο επιθεμάτων όλων αυτών των συμβολοσειρών επιβάλλεται ο περιορισμός για κάθε συμβολοσειρά x_j να υπάρχουν το πολύ $|x_j|$ εσωτερικές ακμές που σχετίζονται με κάποια υποσυμβολοσειρά της x_j . Σε αυτό το δέντρο η διαγραφή γίνεται σε γραμμικό χρόνο και με τη διαγραφή μιας συμβολοσειράς αυτή δεν καταλαμβάνει χώρο στη μνήμη. Έτσι μειώνονται οι χωρικές απαιτήσεις του προβλήματος και οι χρόνοι ενημέρωσης και αναζήτησης παραμένουν σταθεροί. Οι αλγόριθμοι εισαγωγής (ενημέρωσης του S) της συμβολοσειράς στο δέντρο επιθεμάτων και διαγραφής της αντίστοιχα από αυτό αναλύονται στην πρώτη επιστημονική εργασία από τις αναφορές. Η δεύτερη επιστημονική εργασία κάνει μια αναφορά στα τεράστια μεγέθη που απαιτούνται για την αποθήκευση των δένδρων επιθεμάτων αλλά και των πινάκων επιθεμάτων. Άκομη η ανάγκη να παραμένουν συνεχώς στη μνήμη οι παραπάνω δομές δυσκολεύει την ευρεία χρήση τους σε πραγματικές εφαρμογές. Όπως και με την προηγούμενη επιστημονική εργασία βασικό πρόβλημα που προκύπτει από την προσπάθεια δυναμικής διατήρησης μιας συλλογής συμβολοσειρών είναι ο αποθηκευτικός χώρος. Ήως τώρα έχουν παρουσιαστεί μέθοδοι οι οποίοι μειώνουν τον χώρο των δύο παραπάνω δομών συμπλέζοντας τες. Έχουν παρουσιαστεί οι **συμπιεσμένοι πίνακες επιθεμάτων**, το **συμπιεσμένο δέντρο επιθεμάτων** και το **πλήρες συμπιεσμένο δέντρο επιθεμάτων**. Αυτές οι δομές έχουν το μειονέκτημα ότι είναι στατικές. Πρόσφατα σε άλλη επιστημονική εργασία έχει προταθεί η

δυναμική έκδοση του συμπιεσμένου πίνακα επιθεμάτων ωστόσο σε αυτή που μελετάω προτείνεται η δυναμική έκδοση του πλήρους συμπιεσμένου δέντρου επιθεμάτων η οποία έχει ως βάση τον δυναμικό συμπιεσμένο πίνακα επιθεμάτων. Όλες οι παραπάνω δομές δεδομένων είναι σχεδιασμένες ώστε να χειρίζονται συλλογή κειμένων (συμβολοσειρών) όπου οι συμβολοσειρές προστίθενται και διαγράφονται από τη συλλογή. Το **πλήρως συμπιεσμένο δυναμικό δέντρο επιθεμάτων** συνδυάζει τον **συμπιεσμένο δυναμικό πίνακα επιθεμάτων** και το **δυναμικό δ-δειγματοληπτημένο δέντρο (δ-Sampled Tree)**. Τέλος χρησιμοποιείται και ένας αλγόριθμος ο οποίος αντιστοιχίζει τους κόμβους του δ-δέντρου σε στοιχεία του δυναμικού συμπιεσμένου πίνακα επιθεμάτων. Λεπτομέρειες για την δομή και λειτουργία των παραπάνω δομών και κατά συνέπεια των πράξεων εισαγωγής συμβολοσειρών στο σύνολο και διαγραφής τους, βρίσκονται στις αναφορές.

Ερώτημα 4

Αρχικά παρουσιάζω κάποιους βασικούς ορισμούς και έννοιες για το πρόβλημα εύρεσης επαναλήψεων σε κάθε ακολουθία όπου η ακολουθία που επαναλαμβάνεται είναι η **ίδια** σε όλες τις ακολουθίες: Μία **επανάληψη** σε μία συμβολοσειρά είναι μία υποσυμβολοσειρά που εμφανίζεται πολλές φορές μέσα σε αυτή. Η απόσταση ή αλλιώς ο αριθμός των χαρακτήρων μεταξύ των εμφανίσεων αυτής της υποσυμβολοσειράς λέγεται **κενό**. Όταν το κενό αυτό ισούται με μηδέν, δηλαδή οι εμφανίσεις της υποσυμβολοσειράς αυτής βρίσκονται ακριβώς η μία δίπλα στην άλλη, τότε η **επανάληψη** αυτή είναι ένας διαδοχικός πίνακας (tandem array). Ένα διαδοχικό ζευγάρι είναι μία συμβολοσειρά της μορφής: s's' όπου s' μία μη-κενή

συμβολοσειρά. Μία γενίκευση αυτών των διαδοχικών ζευγαριών είναι τα **ζευγάρια**. Ένα ζευγάρι είναι πρακτικά η εμφάνιση της ίδιας υπο-συμβολοσειράς δύο φορές σε μία ακολουθία. Μία συμβολοσειρά s μήκους n είναι μία ακολουθία της μορφής: $s[1..n]=s(1)s(2)\dots s(n)$, όπου $s(i) \in \Sigma$, όπου Σ το αλφάβητο της με $1 \leq i \leq n$. Ένας **παράγοντας** f , μήκους p με θέση εμφάνισης i σε μία συμβολοσειρά s , είναι μία ακολουθία διαδοχικών χαρακτήρων μήκους p , η οποία ξεκινάει από τη θέση i της συμβολοσειράς s και τελειώνει στη θέση $i + p - 1$. Μία **επανάληψη** F πολλαπλότητας m είναι η εμφάνιση ενός παράγοντα f σε μία συμβολοσειρά s , m φορές. Αν i η i -οστή εμφάνιση του παράγοντα f στην συμβολοσειρά και $i+1$ η $(i+1)$ -οστή εμφάνιση του, τότε g_i το κενό μεταξύ τους όπως ορίστηκε παραπάνω. Αν υπάρχουν περιορισμοί στις τιμές που μπορεί να πάρει το κενό αυτό τότε θα ισχύει η σχέση: $d_{\min_i} \leq g_i \leq d_{\max_i}$. Δηλαδή το μέγεθος του κενού φράσσεται μεταξύ ενός κάτω και ενός άνω κατωφλίου. Άρα η F περιγράφεται από το ζεύγος (f, d) όπου f η συμβολοσειρά που επαναλαμβάνεται και d η πλειάδα: $((d_{\min_1}, d_{\max_1}), (d_{\min_2}, d_{\max_2}), \dots (d_{\min_{(m-1)}}, d_{\max_{(m-1)}}))$.

Υπάρχουν δύο βασικά προβλήματα εύρεσης επαναλήψεων σε ακολουθίες: α) Το πρώτο πρόβλημα αφορά την εύρεση επαναλήψεων πολλαπλότητας m σε τουλάχιστον q συμβολοσειρές ενός συνόλου N συμβολοσειρών, μέσου μήκους n , με $q \leq N$ και $m \geq 2$ χωρίς περιορισμούς στο μέγεθος των κενών ανάμεσα στις εμφανίσεις. β) Το δεύτερο πρόβλημα αφορά την εύρεση επαναλήψεων πολλαπλότητας m σε τουλάχιστον q συμβολοσειρές ενός συνόλου N συμβολοσειρών, μέσου μήκους n , με $q \leq N$ και $m \geq 2$, με τη διαφορά από το πρώτο πρόβλημα ότι εδώ πρέπει να ικανοποιείται ο εξής περιορισμός για τα κενά μεταξύ των εμφανίσεων της συμβολοσειράς που επαναλαμβάνεται: Για κάθε i , θα ισχύει ότι $d_{\max_i} - d_{\min_i} \leq c$, όπου c κάποια σταθερά.

Και τα δύο αυτά προβλήματα λύνονται με αλγορίθμους που χρησιμοποιούν το **δέντρο επιθεμάτων** και το **γενικευμένο δέντρο επιθεμάτων**. Το δέντρο επιθεμάτων για μία συμβολοσειρά μήκους n , είναι ένα δέντρο με n φύλλα (όσες και οι καταλήξεις-επιθεμάτων της συμβολοσειράς). Για κάθε φύλλο i του δέντρου επιθεμάτων, αν συνενώσουμε τις ετικέτες των ακμών του

μονοπατιού από τη ρίζα μέχρι το φύλλο αυτό τότε παίρνουμε το επίθεμα της συμβολοσειράς στη θέση i (που ξεκινά από την θέση i). Το γενικευμένο δέντρο επιθεμάτων για ένα σύνολο S από συμβολοσειρές είναι ένα δέντρο επιθεμάτων όλων των συμβολοσειρών που ανήκουν στο S . Το i -οστό φύλλο του γενικευμένου δέντρου επιθεμάτων, είναι το επίθεμα μιας συμβολοσειράς s_j και αναπαρίσταται από το ζεύγος (i, j) όπου δηλώνει την εμφάνιση του συγκεκριμένου επιθέματος στη θέση i της συμβολοσειράς s_j .

Για το πρώτο πρόβλημα (αλλά σε μόνο μία συμβολοσειρά) έχει προταθεί αλγόριθμος ο οποίος βρίσκει όλα τα ζευγάρια σε μία συμβολοσειρά s μήκους n , χωρίς περιορισμούς στα κενά μεταξύ των εμφανίσεων της υπο συμβολοσειράς. Χρησιμοποιεί το δέντρο επιθεμάτων της s . Όταν κατασκευαστεί το δέντρο, διασχίζοντας το δέντρο από τα φύλλα προς τη ρίζα (κάτω προς τα πάνω προσέγγιση), ο αλγόριθμος μπορεί να βρει τα ζευγάρια σε κάθε επίπεδο του δέντρου. Κάθε εσωτερικός κόμβος ν διατηρεί μία λίστα φύλλων, η οποία αποθηκεύει όλα τα επιθέματα της s που λήγουν σε κάποιο φύλλο που βρίσκεται στο υπόδεντρο T_v του κόμβου v . Βάσει του παραπάνω αλγορίθμου έχει προταθεί αλγόριθμος ο οποίος επιλύει το δεύτερο πρόβλημα (αλλά μόνο σε μία συμβολοσειρά) δηλαδή την εύρεση των ζευγαριών στη συμβολοσειρά όταν υπάρχουν περιορισμοί για τα κενά μεταξύ των εμφανίσεων της υπο συμβολοσειράς. Στην ουσία χρησιμοποιείται ο αλγόριθμος που επιλύει το πρώτο πρόβλημα, με τη διαφορά ότι οι λίστες των φύλλων υλοποιούνται ως AVL δέντρα.

Το ζητούμενο πρόβλημα θα επιλυθεί εφόσον μιλάμε για παραπάνω από μία συμβολοσειρά με τη χρήση του **γενικευμένου δέντρου επιθεμάτων**.

Πιο συγκεκριμένα για το πρόβλημα εύρεσης επαναλήψεων μιας υποσυμβολοσειράς s σε παραπάνω από 2 συμβολοσειρές με περιορισμούς στα κενά μεταξύ των εμφανίσεων της συμβολοσειράς που επαναλαμβάνεται, υπάρχει το εξής πρόβλημα. Ενώ στον πρώτο αλγόριθμο που δεν υπάρχουν περιορισμοί στα κενά, ήταν εύκολη η συγχώνευση των λιστών φύλλων δύο κόμβων στο ίδιο επίπεδο του δέντρου με ίδιο κόμβο-πρόγονο εδώ παρουσιάζει ορισμένες δυσκολίες που τα κενά είναι περιορισμένα. Οι λίστες των αδερφών κόμβων πρέπει αρχικά να ταξινομηθούν και μετά να

συγχωνευτούν. Όπως και στον πρώτο αλγόριθμο, ο αλγόριθμος αυτός αποτελείται από 2 στάδια. Στην αρχή του δεύτερου σταδίου ο αλγόριθμος κατασκευάζει τις λίστες φύλλων κάθε εσωτερικού κόμβου από κάτω προς τα πάνω συγχωνεύοντας τις λίστες φύλλων των άμεσων κόμβων παιδιών του αφού πρώτα τις ταξινομήσει. Οι ταξινομημένες λίστες αναπαρίστανται ως finger search trees.

Αναφορές

- Bakalis, A., et al. "Locating Maximal Multirepeats in Multiple Strings Under Various Constraints." *The Computer Journal*, vol. 50, no. 2, 2007, pp. 178-185.
- Choi, Y., and T. W. Lam. "Dynamic suffix tree and two-dimensional texts management." *Information Processing Letters*, vol. 61, 1997, pp. 213-220.
- ScienceDirect*,
<https://www.sciencedirect.com/science/article/pii/S0020019097000185>.
- Russo, Luís M. S, et al. "Dynamic Fully-Compressed Suffix Trees." *Springer*, vol. 5029, 2008, 191--203.
- Stoye, Jens, et al. "Finding Maximal Pairs with Bounded Gap." *Springer Berlin Heidelberg*, 1999, 134--149. *Springer*.

Ερώτημα 5

Ερώτημα 6

Έχω τις ακολουθίες $v = \text{GUGTTGTGG}$ και $w = \text{TCGTGAATT}$ με τις εξής παραδοχές: **α)** Το κόστος στοίχισης είναι +1. **β)** Το κόστος ασυμφωνίας και το κόστος στοίχισης με κενό είναι -1.

- I. Για τον υπολογισμό της ολικής στοίχισης ανάμεσα στις ακολουθίες v και w , ο πίνακας δυναμικού προγραμματισμού διαμορφώνεται ως εξής:

		T	C	G	T	G	A	A	T	T
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
G	-1	-1	-2	-1	-2	-3	-4	-5	-6	-7
U	-2	-2	-2	-2	-2	-3	-4	-5	-6	-7
G	-3	-3	-3	-1	-2	-1	-2	-3	-4	-5
T	-4	-2	-3	-2	0	-1	-2	-3	-2	-3
T	-5	-3	-3	-3	-1	-1	-2	-3	-2	-1
G	-6	-4	-4	-2	-2	0	-1	-2	-3	-2
T	-7	-5	-5	-3	-1	-1	-1	-2	-1	-2
G	-8	-6	-6	-4	-2	0	-1	-2	-2	-2
G	-9	-7	-7	-5	-3	-1	-1	-2	-3	-3

Με την ολική στοίχιση να είναι η :

T C G T - G A A T T -

G U G T T G - - T G G

Με βαθμολογία: -3.

- II. Για τον υπολογισμό της τοπικής στοίχισης ανάμεσα στις ακολουθίες v και w , ο πίνακας δυναμικού προγραμματισμού διαμορφώνεται ως εξής:

		T	C	G	T	G	A	A	T	T
	0	0	0	0	0	0	0	0	0	0
G	0	0	0	1	0	1	0	0	0	0
U	0	0	0	0	0	0	0	0	0	0
G	0	0	0	1	0	1	0	0	0	0
T	0	1	0	0	2	1	0	0	1	1
T	0	1	0	0	1	1	0	0	1	2
G	0	0	0	1	0	2	1	0	0	1
T	0	1	0	0	2	1	1	0	1	1
G	0	0	0	1	1	3	2	1	0	0
G	0	0	0	1	0	2	2	1	0	0

Η τιμή της βέλτιστης τοπικής στοίχισης στις ακολουθίες v και w είναι 3 και αντιστοιχίζεται στη στοίχιση: GTG-GTG. Τα κόκκινα κελιά αντιστοιχούν στην πληροφορία οπισθοδρόμησης.

Ερώτημα 7

Περιβάλλον υλοποίησης και βιβλιοθήκες

Για την υλοποίηση του ερωτήματος χρησιμοποιήθηκε η γλώσσα προγραμματισμού **python** έκδοση **3.10** και συγκεκριμένα οι βιβλιοθήκες:

1. **regex**
2. **Bio (Biopython)**

Πηγαίος Κώδικας

Ο κώδικας που υλοποιεί το παρόν ερώτημα είναι ο εξής:

```
import regex as re
from Bio import SeqIO
from Bio.Seq import Seq
from Bio.SeqIO import parse
from Bio.SeqRecord import SeqRecord
from Bio.SeqUtils import gc_fraction # gia to pososto GC se sxesh me thn
akolouthia

seq = []
records = []
with open('sequence.fasta', 'r') as file:
    # seq = file.read()
    records = parse(file, "fasta")
    for record in records:
        seq = record.seq # einai sequence

# print(seq)
seq = str(seq) # thn kano string
sequence = Seq(seq)
# input('...')
print(sequence)
#####
print('Base count of the sequence: ' + str(len(sequence)) + '\nGc percentage: ' +
+ str(100 * gc_fraction(sequence)) + '%')

#####
RUNX1 = re.finditer('[CGT][ACT]TGTGGT[CT][AT]', seq, overlapped=True)
# BHTGTGGTYW
TGIF1 = re.finditer('[AT]GACAG[CGT]', seq, overlapped=True)
# WGACAGB
IKZF1 = re.finditer('[CGT]TGGGA[AG][AGT]', seq, overlapped=True)
# BTGGGARD
# print('RUNX1 : ', RUNX1)
with open('destination.txt', 'w') as destination:
```

```
for match in RUNX1:
    print(match.group(), "start: ", match.start())
    destination.write(str(match.group()) + " ,position: " +
str(match.start()) + '\n')

for match in TGIF1:
    print(match.group(), "start: ", match.start())
    destination.write(str(match.group()) + " ,position: " +
str(match.start()) + '\n')

for match in IKZF1:
    print(match.group(), "start: ", match.start())
    destination.write(str(match.group()) + " ,position: " +
str(match.start()) + '\n')
    # ousiastika to start() metra posoi xaraktires yparxoun prin. Diladi h arxh
    tis akolouthias prin einai 0 den yparxei kanenas
    # xaraktiras prin, eno to 505 px simenei edo pou einai o kersoras exo 505
    xaraktires prin kai h zhtoumenh ypoakolouthia einai
    # amesos meta

# print('IKZF1 : ', IKZF1)

###

new_sequences = []
new_sequences.append(SeqRecord(sequence.complement(), name='Complementary
Sequence'))
new_sequences.append(SeqRecord(sequence.transcribe(), name='Transcribed
Sequence'))

# input('...')

# grafo ta statistika(arithmos vaseon tis akolouthias kai to gc pososto) tis
akolouthias se ena ksexoristo arxeio
# epishs grafo tis dyo nees akolouthies(th sybliromatikh kai thn metagrafomenh
) se ksexoristo arxeio
with open('statistics.txt', 'w') as statistics:
    statistics.write('Base count of the sequence: ' + str(len(sequence)) + '\nGc
percentage: ' + str(100 * gc_fraction(sequence)) + '%')
with open('complementary-transcription-sequence.fasta', 'w') as sequences:
    SeqIO.write(new_sequences, sequences, 'fasta')
```

Αλγορίθμική λογική της υλοποίησης

Αρχικά εισάγω τις βιβλιοθήκες **Bio** (και κάποιες χρήσιμες υπο-βιβλιοθήκες της) και **regex**. Η επιλογή της **regex** αντί της `re` έγινε γιατί υποστηρίζει επικαλυπτόμενα matches (εμφανίσεις της κανονικής έκφρασης), οπότε αποκλείω την πιθανότητα αποτυχίας εύρεσης κάποιας εμφάνισης κάποιου μεταγραφικού παράγοντα. Αρχικά, αφού έχω αποθηκεύσει την ακολουθία στο αρχείο **sequence.fasta**, διαβάζω με τη μέθοδο **parse()** όλες τις εγγραφές του fasta αρχείου (μόνο την ακολουθία της εκφώνησης) και εξάγω την ακολουθία σε μία μεταβλητή **seq**. Δημιουργώ δύο στιγμιότυπα της ακολουθίας ένα μορφής **string** και ένα μορφής **Seq** για αποδοτικότερο χειρισμό. Αφού τυπώσω την ακολουθία της εκφώνησης, τυπώνει δύο στατιστικά στοιχεία της: Το μήκος της-πλήθος βάσεων της και ποσοστό **gc** σε σχέση με την ακολουθία. Έπειτα με τη χρήση της συνάρτησης **finditer** της **regex** φτιάχνω τις κανονικές εκφράσεις για τους 3 μεταγραφικούς παράγοντες. Η **finditer** παίρνει ως ορίσματα το μοτίβο που αναζητούμε, την ακολουθία στην οποία θα γίνει η αναζήτηση και μία δυαδική μεταβλητή η οποία σηματοδοτεί την εύρεση ή μη των επικαλυπτόμενων εμφανίσεων του μοτίβου. Αυτή ορίζεται σε **true**. Για το μοτίβο κάθε μεταγραφικού παράγοντα εργάζομαι ως εξής: παίρνω το σταθερό κομμάτι κάθε μεταγραφικού παράγοντα (δηλαδή τις βάσεις **A,C,G,T**) και το τοποθετώ στο μοτίβο. Για το μη σταθερό κομμάτι του μοτίβου δηλαδή τους κώδικες ασάφειας χρησιμοποιώ τον τελεστή `[]` μιας κανονικής έκφρασης στην **regex**. Ο τελεστής `[]` δηλώνει μια κλάση χαρακτήρων και επιστρέφει ταίριασμα εάν είναι παρών στο κείμενο κάποιος χαρακτήρας της κλάσης αυτής. Για παράδειγμα, για τον κώδικα ασάφειας **B** η κλάση χαρακτήρων του θα είναι η `[CGT]` την οποία προσθέτω στο μοτίβο του μεταγραφικού παράγοντα που ανήκει. Έπειτα για κάθε αντικείμενο κάθε μεταγραφικού παράγοντα που εντοπίστηκε στην ακολουθία γράφω στο αρχείο **destination.txt** τον παράγοντα που εμφανίστηκε και τη θέση του στην ακολουθία. Έπειτα δημιουργώ δύο εγγραφές μέσω της **SeqRecord()** η οποίες περιέχουν την ακολουθία μετά από μεταγραφή και την συμπληρωματική ακολουθία της αρχικής ακολουθίας. Αυτό το κάνω μέσω των συναρτήσεων **complement()** και **transcribe()**. Τέλος γράφω τις δύο αυτές εγγραφές στο

αρχείο **complementary-transcription-sequence.fasta** και τα στατιστικά που εντόπισα στο **statistics.txt**.

Αποτελέσματα

Τρέχοντας το πρόγραμμα τυπώνονται στην οθόνη τα εξής αποτελέσματα:

```
C:\Users\Vergossa\PycharmProjects\Bio2024\venv\Scripts\python.exe C:\Users\Vergossa\PycharmProjects\Bio2024\erotima7.py
GACACCTAGTACTA66ATGATCACCTGAACTAGCAGGCCCTGGTTCCAATTTCATCAACACTGTA66666ATTATCCTA6A66668TC666ATTTCATCAGAGTATTTCCTGCTGCTCTTACAATTGGGAACAATAATTGAGTGGTTATTACCCCTGGCTACGCACTGGAAACTTTAAAATAATGCTGGTATGAAATTACAC
Base count of the sequence: 1397
Gc percentage: 40.873299928418035%
CATGTGGTTA start: 252
AGACAGC start: 297
AGACAGC start: 1037
TTGGGAAG start: 448
GTGGGAAT start: 554
CTGGGAGT start: 791
Process finished with exit code 0
```

Και τα περιεχόμενα των αρχείων που ανέφερα στην προηγούμενη παράγραφο είναι τα εξής:

destination.txt

```
CATGTGGTTA ,position: 252
AGACAGC ,position: 297
AGACAGC ,position: 1037
TTGGGAAG ,position: 448
GTGGGAAT ,position: 554
CTGGGAGT ,position: 791
```

statistics.txt

```
base count of the sequence: 1397
Gc percentage: 40.873299928418035%
```

complementary-transcription-sequence.fasta

```

1  ><unknown id> <unknown description>
2  CTGTGGAGTCATGATCCTACATAGTCGGACTTGTACGTCCGGACCAAGGTTAAAAAAAT
3  AGTTGTGAGCATCCCCCTAATAGGATCTCCCCAGACCCCTAAAGAAACTGTAGTCTCATA
4  AAAACGGAACGAGGAAGTGTAAACCCCTTGTATTAAATCACCAATAATTGGGACCGAT
5  GCGTGACCTTGAAATTTTATTACGACCATACTTAAATGTGTCTCATAGCACCTTAA
6  AAGTGAECTCATGGTACACCAATATGTAACCTATTCCGAGGTCTTCGTCATGACCTTCT
7  GTCGGTACGGTTCTCACCAATCACCAACCTAAAACCGTTAGTCAGTCAAAATCAGACGGAAT
8  AGTTTATGTACCCGTATGTCTATTAGGAATCTACCGAGAGGATGAATGACTTTGTAAAA
9  GATAGATAGATAGATAGATAGATAAACCTTCGATAGATAGATAGATAGATAAATAA
10 ATTCCATCAGAGATAGACGGAGACAGAGACAGACAGAGACAGAGACACAGACAGACGA
11 GAGAGAGAGAGAGACACCCTTAGAGAGAGACACACACACACACACACACACACACA
12 CACACCACACGTACTTGTACTCATTTAGGTATTCTTGTAAAGTCTCAACCAGGAGAGG
13 AATATAGTTACCTAGGTCTTAATTGAGTCAGTTAAGAACCCACGGAAATGATCAAC
14 TCGGTAGAGTGACCGAGAAGTAGTAGAAATCTTATTGAGTCAAATAATGTGTGTGTG
15 TGTGTGTGTTGGACCCCTCATGTGTGTGTGTGGTTGGGGTTGCCTTTGATGTT
16 ATAATATTACTTATGTGTCCAAGAGTTGTATCAGAGACGGTGCACGTCTGTTACT
17 CATCTTCATCTTCTTGGCCCTTGCACCTCGTTAGTCAGTCTCCTTATTGTCAGTCTC
18 TTATTGTCAGTCTCCTTATTGTCAGTCTCCTCATTGTCAGTCTCCTTATGTCAGTC
19 TTCCTTATTGTCAGTCTCTGTCGTGTCAGTCCTCTTATTGTCAGTCTCCTTATTGTC
20 AGTCCTCCTTATTGTCAGTCTCCTTATTGTCAGTCAGTCAGTCAGTCAGTCAGTCCTTAT
21 TGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTC
22 TATCGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTC
23 TATTGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTCAGTC
24 GTTTGTGTCTCTACTGTTCCGTTACCCCAGTCAGTCAGTCAGTCAGTCAGTCAGTC
25 TATATGAGAGAGACACA
26 ><unknown id> <unknown description>
27 GACACCUUCAGUACUAGGAUGUAUCAGCCUGAACUAGCAGGCCUGGUUCCAAUUUUUUUA
28 UCAACACUCGUAGGGGGAUUAUCCUAGAGGGGGUCUGGGAUUUCUUUGACAUCAAGAGUAU
29 UUUUGCCUUGCUCCUUCACAAUUUUGGAACAAUUAUUUAGUGGUUUAUACCUGGCUA
30 CGCACUGGAAACUUUAAAAAUUGCUGGUUAGAAUUAACACAGAGUAUCGUGAAAAAU
31 UUCACUGAGUACCAUGUGGUUAUACAUUUGGAAGGGCUCAGGAAGCAGCUACUGGAAGA
32 CAGCCAUGCCAAGAGUGGUUAGUGGUUAGGGCAAGUCAGCUUUUAGUCUGCCUUA
33 UCAAAUACAUGGGCAUACAGAUAAAUCUUAAGAUGGCUCCUACUUACUGAAACAUUUU
34 CUAUCUAUCUAUCUAUCUAUCUAUUCUAAUUGGAAGCUAUCUAUCUAUCUAUCAUUUUA
35 UAAGGUAGUCUCAUCUGCCUCUGUCUGUCUGUCUGUCUGUCUGUCUGUCUGUCUGCU
36 CUCUCUCUCUGUGGGAAUCUCUCUGUGUGUGUGUGUAUGUGUGUGUGUGUGUGUGU
37 GUGUGGUGUGCAUGAACAGAGUAAAUCUAAAGGAAACUUUCAGAGUUGGUCCUCUCC
38 UUUAUCAAAUGGAUCCAGGAAUAAAUCUAGGUUCAUUCUUGGUGCCUUUACUAGUUG
39 AGCCAUCUCACUGGCUCUCAUCAUCUUAGAAUAAUCACACACACACACACACACAC
40 ACACACACAACCUGGGAGUACACACACACACACACACACACACACACACACACAC
41 UAUUAUAUGAAUACACAGGUUCUCAACAUAGUCUCUGCCACGCUUGCAGACAAAGAUGA

```

42	GUAGAAGUAGAAAGAACCAAGGGAAACGUGGAGCAAGUCAGAAGGAAUAACAGUCAGAAGG
43	AAUAAACAGUCAGAAGGAAUAACAGUCAGAAGGAGUAACAGUCAGAAGGAAUAAGCAGUCAG
44	AAGGAAUAACAGUCAGAAGACAGCACAGUCAGAAGGAAUAACAGUCAGAAGGAAUAACAG
45	UCAGAAGGAAUAACAGUCAGAAGGAAUAACAGUCAGAAGGAAUAAGCAGUCAGAAGGAAUA
46	ACAGUCAGAAGGAAUAACAGUCAGAAGGAAUAACAGUCAAAGGAAUAAGCAGUCAGAAGGAA
47	AUAGCAGUCAGAAGGAAUAACAGUCAAAGGAGCAGUCAGAAGGAGUAACAGUCAGAAGGAA
48	AUAACAGUCAGAAGGAAUAACAGUCAAAGGAAUAAGCAGUCAGAAGGAGUAACAGUCAGAAG
49	CAAACACAGAGAUGACAAGGCAAUGGGGUCAAGACACUUUACCACUCUCCAAGAUCAUCUAC
50	<u>AUAUACUCUCUCUGUGU</u>
51	