



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

Τμήμα Μηχανικών Η/Υ και Πληροφορικής

Εισαγωγή στη Βιοπληροφορική - Προπτυχιακό  
(CEID1047)

Πρώτο Σύνολο Ασκήσεων 2021-2022

Στεφανίδης Μάριος — 1067458  
Μητροπούλου Αικατερίνα — 1067409

Πάτρα, Ιούνιος 2022

## 1 Άσκηση Πρώτη

## 1.1 Introduction to the Bioinformatics Armory

Πιο συγκεκριμένα, δίνεται ως είσοδος μια ακολουθία DNA (=S) που έχει μήκος το πολύ 1000-bp και επιστρέφονται 4 ακέραιοι αριθμοί (χωρισμένοι με κενά) που αντιπροσωπεύουν το πλήθος των φορών που εμφανίστηκαν τα σύμβολα 'A', 'C', 'G' και 'T' στο S.

Εικόνα 1: Είσοδος

Εικόνα 2: Έξοδος

<sup>1</sup> Αποφύγαμε να κατεβάσουμε τοπικά στους υπολογιστές μας τα λογισμικά των προτεινόμενων εργαλείων και προσπαθήσαμε να χρησιμοποιήσουμε αποκλειστικά τις online ηλεκτρονικές διεπαφές, όπου αυτές ήταν διαθέσιμες

| Βάση         | Πλήθος |
|--------------|--------|
| Αδερίνη (A)  | 193    |
| Θυμίνη (T)   | 212    |
| Κυτοσίνη (C) | 213    |
| Γουανίνη (G) | 210    |

## 1.2 GenBank Introduction

Με τη χρήση του συγκεκριμένου λογισμικού δίνεται η δυνατότητα να κάνουμε απλά ερωτήματα με χρήση περιορισμών στην [GenBank](#).

Πιο συγκεκριμένα, μπορούμε να πραγματοποιήσουμε αναζήτηση στη βάση Δεδομένων Nucleotide με γενικά ερωτήματα κειμένου, η οποία θα παράγει στη συνέχεια τα πιο σχετικά αποτελέσματα. Δίνεται ακόμα η δυνατότητα να κάνουμε ένα απλό ερώτημα που βασίζεται σε όνομα πρωτεΐνης, όνομα γονιδίου ή σύμβολο γονιδίου. Ωστόσο, σε περίπτωση που επιθυμούμε να περιορίσουμε την αναζήτηση μας σε συγκεκριμένα είδη εγγραφών, μπορούμε να χρησιμοποιήσουμε την σελίδα Limits page της GenBank ή εναλλακτικά το πεδίο φιλτραρίσματος των αποτελεσμάτων για να επιλέξουμε συγκεκριμένες κατηγορίες εγγραφών.

Παραδείγματος χάριν, δίνεται στο λογισμικό ως είσοδος το όνομα γένους, ακολουθούμενο από δύο ημερομηνίες σε μορφή EEEE/M/H και επιστρέφεται ο αριθμός των εγγραφών Nucleotide GenBank για το συγκεκριμένο γένος που δημοσιεύτηκε μεταξύ των καθορισμένων ημερομηνιών.

### Nucleotide Advanced Search Builder

(Anthoxanthum) AND ("2003/7/25"[Publication Date] : "2005/12/27"[Publication Date])

Edit Clear

Builder

All Fields Anthoxanthum Show index list

AND Publication Date 2003/7/25 to 2005/12/27 Show index list

AND All Fields Show index list

Search or Add to history

History

There is no recent history

Εικόνα 3: Αναζήτηση

Όπως φαίνεται και παραπάνω, έχει χρησιμοποιηθεί το γραφικό περιβάλλον της GenBank και έχει πραγματοποιηθεί η αναζήτηση του Anthoxanthum μεταξύ των ημερομηνιών 2003/7/25 έως 2005/12/27 (εικόνα 3). Η παραπάνω αναζήτηση επιστρέφει 54 αποτελέσματα (εικόνα 4).

## 1.3 Data Formats

Στο συγκεκριμένο πρόβλημα δίνουμε περισσότερη έμφαση στο Format που χρησιμοποιούν οι απαντήσεις που επιστρέφονται από τα ερωτήματα στη GenBank (βλ. ενότητα 1.2). Ακόμα, ερχόμαστε σε επαφή με το εργαλείο GenBank to FASTA.

Δίνεται ως είσοδος μια συλλογή από n ( $n \leq 10$ ) αναγνωριστικά εισόδου GenBank (εικόνα 5) και επιστρέφεται η συντομότερη από τις συμβολοσειρές που σχετίζονται με τα αναγνωριστικά σε μορφή FASTA (εικόνα 6).

Items: 1 to 20 of 54

<< First < Prev Page 1 of 3 Next > Last >>

- ☐ [Anthoxanthum odoratum chloroplast partial rbcL gene for ribulose biphosphate carboxylase large chain](#)  
1. 1,408 bp linear DNA  
Accession: AJ746282.1 GI: 57283843  
[Protein](#) [Taxonomy](#)  
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum Rec A1 chloroplast microsatellite sequence](#)  
2. 113 bp linear DNA  
Accession: AY243051.1 GI: 33413983  
[Taxonomy](#)  
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum 1973 Aa chloroplast microsatellite sequence](#)  
3. 107 bp linear DNA  
Accession: AY243050.1 GI: 33413982  
[Taxonomy](#)  
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum 1925 A1 chloroplast microsatellite sequence](#)  
4. 105 bp linear DNA  
Accession: AY243049.1 GI: 33413981  
[Taxonomy](#)  
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum RC16 1900 A2/1925 A2 chloroplast microsatellite sequence](#)  
5. 113 bp linear DNA  
Accession: AY243048.1 GI: 33413980  
[Taxonomy](#)  
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum chloroplast partial rbcL gene for ribulose biphosphate carboxylase large chain](#)  
6. 1,428 bp linear DNA  
Accession: AJ746256.1 GI: 57283791  
[Protein](#) [Taxonomy](#)  
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Hierochloa odorata chloroplast partial rbcL gene for ribulose biphosphate carboxylase large chain](#)  
7. 1,358 bp linear DNA

Εικόνα 4: Αποτελέσματα

## 1.4 New Motif Discovery

Με το συγκεκριμένο εργαλείο μπορούμε να βρίσκουμε μοτίβα μέσα σε συμβολοσειρές. Αφού δώσουμε την είσοδο (είτε σε μορφή αρχείου είτε συμπληρώνοντας το αντίστοιχο πλαίσιο κειμένου) (εικόνα 7), το MEME εκτελεί ένα "job". Στη συνέχεια, προκύπτουν τα αποτελέσματα, τα οποία μπορούμε να δούμε σε διαφορετικές μορφές (εικόνα 8).

Δίνεται ως είσοδος ένα σύνολο ακολουθιών πρωτεΐνης σε μορφή FASTA που μοιράζονται κάποιο μοτίβο με ελάχιστο μήκος 20 και επιστρέφεται μια κανονική έκφραση για το μοτίβο με την καλύτερη βαθμολογία. Ενδεικτικά, μπορείτε να δείτε τις μορφές MEME HTML Output και MAST HTML Output.

```

ORGANISM      Eukaryota; Metazoa; Echinodermata; Echinozoa; Echinoidea;
              Eueskadozoa; Echinodermata; Echinozoa; Echinoidea;
              Echinodermata; Echinozoa; Echinoidea;
REFERENCE     1 (bases 1 to 2320)
AUTHORS       Kane,R.E.
TITLE         Actin polymerization and interaction with other proteins in
              temperature-induced gelation of sea urchin egg extracts
JOURNAL       J. Cell Biol. 71 (3), 704-714 (1976)
MEDLINE       77051438
REFERENCE     2 (bases 1 to 2320)
AUTHORS       Bryan,J. and Kane,R.E.
TITLE         Separation and interaction of the major components of sea urchin
              actin gel
JOURNAL       J. Mol. Biol. 125 (2), 207-224 (1978)
MEDLINE       79091184
REFERENCE     3 (bases 1 to 2320)
AUTHORS       Bryan,J., Edwards,R., Matsudaira,P., Otto,J. and Mufkuhle,J.
TITLE         Fascin, an echinoid actin-bundling protein, is a homolog of the
              Drosophila singed gene product
JOURNAL       Proc. Natl. Acad. Sci. U.S.A. 90 (19), 9115-9119 (1993)
MEDLINE       94022326
FEATURES
  source       1..2320
              /organism="Strongylocentrotus purpuratus"
              /db_xref="taxon:7668"
              /db_stage="larva"
  gene         1..2320
              /gene="FSCN1"
  CDS          85..1575
              /gene="FSCN1"
              /function="actin bundling protein"
              /note="putative"
              /codon_start=1
              /product="fascin"
              /protein_id="AAC37183.1"
              /db_xref="GI:161471"
              /translation="MDAMNLKYKGLVNSAGRYLTAEKGGKVNASGATLKAROVNITL
              EDEESSTISYLVKAPSONELSAQKNGVAVCSVEDRTEDQDTGFELELPDQKWLKQVS
              HORYLACNGEELICSESSISNPSANMTVQALHPQVCMKNVOHORYAHLKTEEGEDS
              VVDDELVPNGADSTLTLYLGGKYGLEAFNGKEVOTDGLAGTANEOTETLTETSG
              HLVLBNNGBHLGVDSGTRVLKSKRGLTKANYETLEDSCPGAFEEGGKYASLKQGE
              DVSFKLLVDEDTEDTETFOLEFVETDKYAIRVCDPKKNSRDAKFMTVAAGIQANGNS
              KDQTDQCFSEVYNGNDMHVRAPGGKYVSVBRNGHLEQSEPKDFIRLLNRPKLVKCC
              PHGFVGKNGKAEVACNRSNEDVETVITYKEGGYTIQDSCGKYNSCDSSRVLVGEAAG
              TFFFEFHLSKFAIRAESNGMLKGEQSGLEFANGSEVSKDTLWEE"
  polyA_signal 2155..2160
              /gene="FSCN1"
ORIGIN
1  acttgaagat gaataaaatc gactgatacc aaaaacacat tattttaccg aantgatcat
61  ttgaagacat caacataatt cacatgacct gctatgaatt taataacaa atttgacctg
121  gtcaactcag ccaacagata cctcactgct gaagaatttg atnccaaagt caatgacctc
181  ggaacacat taataacgag ccaagatagc atcctagagc aaaaagaaag caaacacatc
241  aactacttga aagcagccctc taataacttc ctctctgacg ataaaaacag taacgctcat
301  tcaactgtag aagacagaac aagagacagc gatacaggaat tcaagatcag attgcaaccc
361  gaatntaat gaacccctca gaattttct caccagaagt acctagcttg caatgatgag

```

Submit Clear Reset

Εικόνα 5: Αναγνωριστικά Εισόδου GenBank

## 1.5 Pairwise Global Alignment

Σε αυτό το σημείο ασχολούμαστε με ένα εργαλείο ολικής στοίχισης ακολουθιών RNA και DNA, το Needle. Δέχεται ως είσοδο δύο ταυτότητες GenBank (εικόνα 9) και επιστρέφεται η μέγιστη συνολική βαθμολογία ευθυγράμμισης μεταξύ των συμβολοσειρών DNA που σχετίζονται με αυτά τα αναγνωριστικά (εικόνα 10).

## GenBank to FASTA results

```
>Strongylocentrotus purpuratus fascin (FSCN1) mRNA, complete cds.
acttgaagtgataaaatcgactgtatccaaaacaacattgttttacagaagtggctgt
ttgaggacatcaacatatttcacaatgccctgtatgaattttaaatacaaaatttggcctg
gtcaactcggccggcagatacctcactgtgagaagtttggtggcaaatgcaatgcctca
ggagcaacgttaaaagccaggcaagtatggatcctagagcaagaagagagcagcacgac
agctacttgaaggcgccctctggtaacttccctctctgcagataaaaaacggtaacgtctat
tgagtggttggagcagagcggaggacgggatacaggattcgagatcgagttgcaaccc
gatggtaaatggggccctcaagaatgtttctcaccagaggtacctagcttgcattggtgag
gagctgatctgcagtggaatccagcaccagcaacccctcagcaaacctggactgtccagctg
gccatccatccacaggctctgatgaagaacgtccagcaccacacgtacgcacatctcaaa
accagtggaggagggtgaagacagcgtggttgtagacgaattggttccctggggagctgat
tcacactcactcttctacctgggcaaaaggaagtagggccttgaggccttcaacgga
aagtttgtccaaacccgacggacagcttctggtggcacaagcaacagacgcagttcaca
ctcatcttcacatccggctcacttggtagtaagggacaacaaatggagctcacttaggagtg
ccagtggaaccagggtcttgaagtccccaagcctggactgacgaagccaattacttc
atcttagaggatagctgtccacaaggtgcttccgaatttgggtggcaaatatgcacgtta
aagcaaggcgaagatgtttcattcaagctctctgtgacgaagatatacgaagacacagag
acctccagttggagtctgttgaaccgacaagtagccatcagggtatgtgaccccaag
aagaactccagagatgctaagtcttgaagacgcgtgctgctggtatccaggctaacggc
aactcaaaaggaccagcagcagctgtcaattctctgtcgaatacaacggcaacgacatgcat
gtgcgtgctccaggaggcaagtagttagtgcgtgacaacggccatctcttccctcag
gattcaccctaaagacttcatcttccgtctgtcaccgacccaagctggtgctcaagtgc
cctcatggattctgtgggcatgaaggaggggcaaggctgaggtgcgctgcaacccgatcaaac
tttgatgtcttcaactgtcacttacaaggaaggcggatatacactataccaagactcctgtggc
aagtactggtcttctgatgacagtagccgcatcgttcttggagaggcagcaggtacttct
tcttctcagattccatgagctctccaagtcttctgatacagagcagaagcaacggcagtggtg
atcaaggcgagcagagtggtcttcttaccgccaatggttccgagggtctcaaaaggacaca
ctgtgggaattctaaacaaattgggcttgaagaagccaaatccaaatcagaagtagagt
agctgacaagccagccactctatctattatatacaattgcaaaatttgtacatttttttaa
tacaaaaaatttttcaaaagtgataataatttcttcaactctggtgggactctttagga
tcatttttctcattgccttggcactgacttccattatccctcattttttaaaggta
aattgatcacttaatacaactgaaaacgaatggaagtaggtctctggaatttagaag
aatagatgactatccagatattcaaatattgttgaacctgtctcaaaaaacccata
aaaaaacctctgttttctgctgtctcagccataaaatagagatcaattctggtggtata
tgctacttaaaatcaggcttgaataagaaataatggaatggaatggatttcaaaaagat
ttggaattttaaatttgcagccagtgctcgtgacaactctgcataccagaagcactaaa
cagctctctgcgcacgcgtcgcacaggtgattgtgtgactgacttcttgaacaaataaa
acagatcttctcgtcaagtgttgataataaagtggattgaaatgcacgaatagattcgac
ttgtatagggcatggtggacattgattttacagatactttcaataaccggtaaaaaatca
atcatatagaaaaatgaaaacagggtgtaatactcaata
```

Εικόνα 6: Συντομότερη Συμβολοσειρά

Perform motif discovery on DNA, RNA, protein or custom alphabet datasets.

Select the motif discovery mode

☒ Classic mode
☐ Discriminative mode
☐ Differential Enrichment mode

Select the sequence alphabet

Use sequences with a standard alphabet or specify a custom alphabet.

☒ DNA, RNA or Protein
☐ Custom

Input the primary sequences

Enter sequences in which you want to find motifs.

Type in sequences

7142

SMRACRVEDLMCKIPVHNDPFLKTVSPAAGHRGMOFDHNFVYPMGO  
YAPARRKRFMCQFFILTFPHFCFRAHSMVENCLPTTVSOFDCTCAIFE  
FRKQHRHOSYPPONGLPNFHNTISWYQKROHICHMAEVLGPVPHPFM  
SRCEENKQNEVIMCKNTSDIKKDGPEIMMYNLPYLTATRIGLRLLALY  
.4494  
FPVLNTHFQEHMKERHNYLDALCHPEYLDGEKXYFNLKQOISCEYR  
GFGKQHFTRTEFHTFRADNTWPCFTHQAEIKIFDEGTSKLYMADF  
GCVGFEDSOYVENQENREYCCGLKSKFQYEHOLMCKIPVHNDPFLK  
RGMOFDHNFSTKCSKSNKCPPOQCGOYLTSVCKCPEYEVYTKREEM  
.3636  
YCRGPLLNDGGYGLPLNDGGYTTISYQAEAFPLRKIFMFMKIDGHSFC  
QHALLGNFMNDTCFHPNLTNLQVPRIVEFAKLIKKEFOMNWCAPDPPA  
NCFHNCFRQVQNDLMCKIPVHNDPFLKTVSPAAGHRGMOFDHNFQPM  
FSCNNVGMQNDVNDYDARAHPFTYIREYADTWSDTSNFRGRIGOM

Select the site distribution

How do you expect motif sites to be distributed in sequences?

Select the number of motifs

How many motifs should MEME find?

Input job details

(Optional) Enter your email address.

(Optional) Enter a job description.

Note: if the combined form inputs exceed 80MB the job will be rejected.

Εικόνα 7: Είσοδος σε μορφή FASTA

Your MEME job is complete. The results should be displayed below.

#### Job Details ...

#### Results

- [MEME HTML output](#)
- [MEME XML output](#)
- [MEME text output](#)
- [MAST HTML output](#)
- [MAST XML output](#)
- [MAST text output](#)
- [\(Primary\) Sequences](#)

#### Status Messages

- Arguments ok
- Starting meme  
meme sequences.fa -protein -oc . -nostatus -time 14400 -mod zoops -nmotifs 3 -minw 6 -maxw 50 -objfun classic -markov\_order 0
- meme ran successfully in 1.07 seconds
- Starting mast  
mast meme.xml sequences.fa -oc . -nostatus
- mast ran successfully in 0.18 seconds
- Done

### Εικόνα 8: Μορφές αποτελέσματος

#### Pairwise Sequence Alignment

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

STEP 1: Enter your nucleotide sequences

Enter a pair of

DNA

sequences. Enter or paste your first **nucleotide** sequence in any supported format:

MSLIGSISNEVPEHPGSPVS

Or, upload a file: [Browse...](#) No file selected. [Use a example sequence](#) [Clear sequence](#) [See more example inputs](#)

AND

Enter or paste your second **nucleotide** sequence in any supported format:

MSNICTNSGNTNNNNVNSKT

Or, upload a file: [Browse...](#) No file selected.

### Εικόνα 9: Είσοδος ακολουθιών DNA

Alignment Submission Details

View Alignment File

```
#####
# Program: needle
# Rundate: Wed 15 Jun 2022 17:27:00
# Commandline: needle
# -auto
# -stdout
# -asequence emboss_needle-I20220615-172657-0811-74431970-p2m.asequence
# -bsequence emboss_needle-I20220615-172657-0811-74431970-p2m.bsequence
# -datafile EDNAFULL
# -gapopen 10.0
# -gapextend 0.5
# -endopen 10.0
# -endextend 0.5
# -aformat3 pair
# -snucleotide1
# -snucleotide2
# Align format: pair
# Report file: stdout
#####

#=====
#
# Aligned sequences: 2
# 1: EMBOS_001
# 2: EMBOS_001
# Matrix: EDNAFULL
# Gap_penalty: 10.0
# Extend_penalty: 0.5
#
# Length: 41
# Identity:      0/41 ( 0.0%)
# Similarity:   0/41 ( 0.0%)
# Gaps:         41/41 (100.0%)
# Score: 0.0
#
#=====

EMBOS_001      1 MSLIGSISNEVPEHPGSPVS----- 21
EMBOS_001      1 -----MSNICTNSGNTNNNNVNSKT 20

#-----
#-----
```

### Εικόνα 10: Μέγιστη βαθμολογία ευθυγράμμισης

## 1.6 FASTQ format introduction

Το FASTQ είναι άλλο ένα format για αναπαράσταση βιολογικών ακολουθιών. Στη συγκεκριμένη περίπτωση, ερχόμαστε σε επαφή με εργαλεία που αφορούν στην μετατροπή του FASRQ σε FASTA.

Τέτοια εργαλεία για παράδειγμα είναι τα εξής:

- Online sequence conversion tool (εικόνα 11)
- Galaxy
- BlastStation

Όπως γίνεται προφανές, δέχεται ως είσοδο ένα αρχείο FASTQ και επιστρέφει τις αντίστοιχες εγγραφές σε μορφή FASTA.

| Format   | About format   |
|----------|--|
| abi      | Reads the ABI "Sanger" capillary sequence traces files, including the PHRED quality scores for the base calls. This allows ABI to FASTQ conversion. Note each ABI file contains one and only one sequence (so there is no point in indexing the file). |
| abi-trim | Same as "abi" but with quality trimming with Mott's algorithm.   |
| ace      | Reads the contig sequences from an ACE assembly file. Uses Bio.Sequencing.Ace internally clustal The alignment format of Clustal X and Clustal W. See also the Bio.Clustalw module.  |
| cif-atom | Uses Bio.PDB.MMCIFParser to determine the (partial) protein sequence as it appears in the structure based on the atomic coordinates.   |

Εικόνα 11: Sequence conversion tool

## 1.7 Read Quality Distribution

Σε αυτό το σημείο ασχολούμαστε με τον έλεγχο ποιότητας. Πιο συγκεκριμένα, υπάρχουν εργαλεία, τα οποία ελέγχουν την ποιότητα της ακολουθίας που έχει δοθεί ως είσοδο, προκειμένου να δούμε αν θα εξάγουμε από την παραπάνω σωστά δεδομένα.

Δίνεται ως είσοδος ένα όριο ποιότητας μαζί με καταχωρήσεις FASTQ για πολλαπλές αναγνώσεις και επιστρέφεται ο αριθμός των αναγνώσεων, των οποίων η μέση ποιότητα είναι κάτω από το όριο. Υπάρχουν οι εξής επιλογές:

- Μια έκδοση του FastQC μπορεί να ληφθεί από το Babraham Bioinformatics και να εκτελεστεί τοπικά σε οποιοδήποτε λειτουργικό σύστημα με εγκατεστημένο κατάλληλο Java Runtime Environment (JRE).
- Μια ηλεκτρονική έκδοση του FastQC είναι επίσης διαθέσιμη στο Galaxy

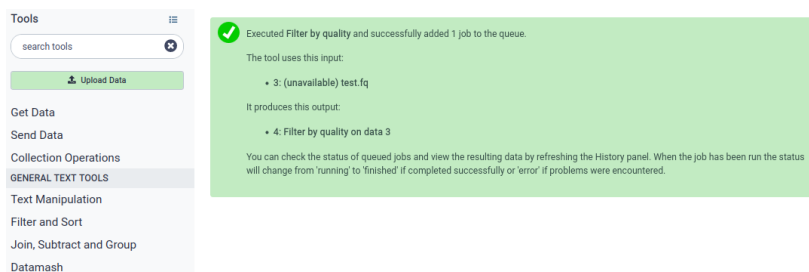
## 1.8 Protein Translation

Το πακέτο SMS 2 (έχει αναφερθεί και παραπάνω) διαθέτει ένα εργαλείο μετάφρασης - **Translate** -, το οποίο μπορεί να μετατρέψει μια αλληλουχία βάσεων σε αλληλουχία αμινοξέων.

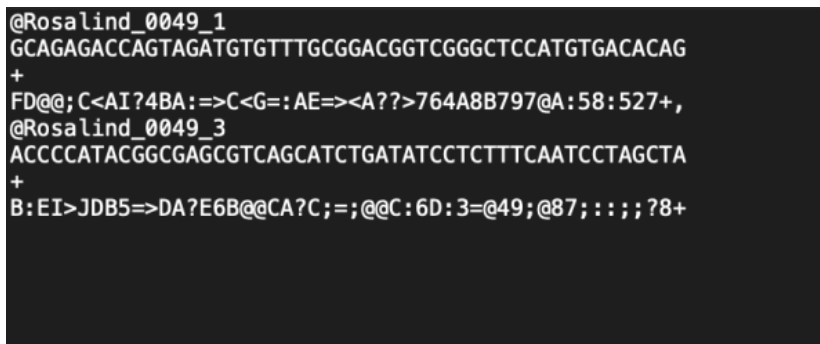
Δέχεται ως είσοδο μια συμβολοσειρά DNA (=S) μήκους το πολύ 10 kbp και μια συμβολοσειρά πρωτεΐνης μεταφρασμένη από την S και επιστρέφει τον δείκτη παραλλαγής γενετικού κώδικα που χρησιμοποιήθηκε για την μετάφραση.







Εικόνα 15: Μήνυμα Επιτυχίας

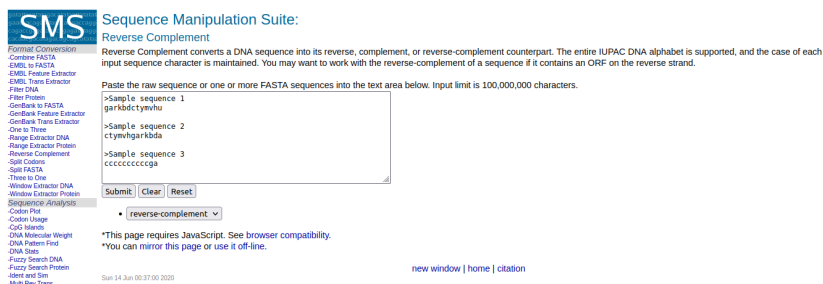


Εικόνα 16: Έξοδος/Αποτέλεσμα

## 1.10 Complementing a Strand of DNA

Γνωρίζουμε πως σε μια ακολουθία DNA (έστω S), τα 'A' και 'T' είναι συμπληρωματικά μεταξύ τους, όπως και τα 'C' και 'G'. Το εργαλείο Reverse Complement του SMS 2 πακέτου λοιπόν υπολογίζει το αντίστροφο συμπλήρωμα μιας αλυσίδας (π.χ., το αντίστροφο συμπλήρωμα του "GTCA" είναι "TGAC").

Δίνεται ως είσοδος μια συλλογή από  $n$  ( $n \leq 10$ ) συμβολοσειρές DNA και επιστρέφεται ο αριθμός των συμβολοσειρών που δόθηκαν ως είσοδος που ταιριάζουν με τα αντίστροφα συμπληρώματά τους (εικόνα 18). Έχει χρησιμοποιηθεί η [online](#) ηλεκτρονική διεπαφή του προγράμματος Reverse Complement (εικόνα 17).

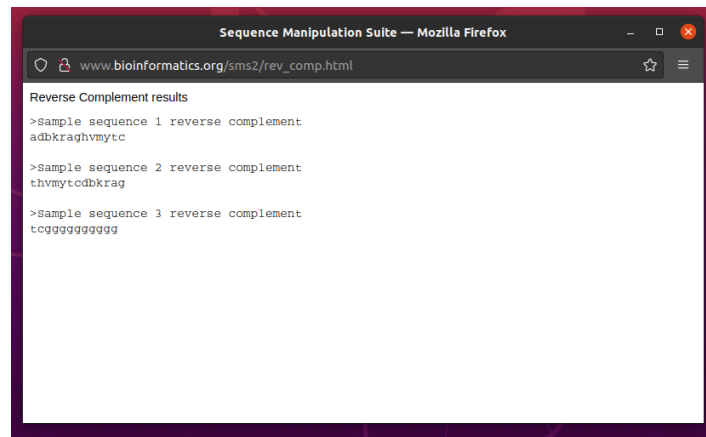


Εικόνα 17: Είσοδος στο Reverse Complement

## 1.11 Suboptimal Local Alignment

Μέσω του εργαλείου Lalign μπορούμε να βρούμε, μεταξύ 2 ακολουθιών, τα πολλαπλά εναλλακτικά τοπικά ταιριάσματα τους.

Δίνονται ως είσοδο 2 συμβολοσειρές DNA S και T σε μορφή FASTA που μοιράζονται κάποια σύντομη ανακριβή επανάληψη  $r$  των 32-40 bp (με τον όρο ανακριβή εννοούμε πως το  $r$  μπορεί να εμφανίζεται με



Εικόνα 18: Μήνυμα Αποτελέσματος

μικρές τροποποιήσεις) (εικόνα 19) και επιστρέφεται ο συνολικός αριθμός εμφανίσεων του  $r$  ως υποσυμβολοσειρά του  $s$ , ακολουθούμενη από τον συνολικό αριθμό εμφανίσεων του  $r$  ως υποσυμβολοσειρά του  $t$ .

Χρησιμοποιείται η online διαθέσιμη ηλεκτρονική διεπαφή. Μπορείτε να βρείτε εδώ τα αποτελέσματα.

## Pairwise Sequence Alignment

LALIGN finds internal duplications by calculating non-intersecting local alignments of protein or nucleotide sequences.

**STEP 1 - Enter your nucleotide sequences**

Enter a pair of

DNA

sequences. Enter or paste your first nucleotide sequence in any supported format:

```
>Rosalind_12
GACTCCTTGTGTCCTAAATAGATACATATTTACTCTTGACTCTTTGTTGGCCTTAAATAGATACATATTTGTGCGACTCCACGAGTGATTCGTA
```

Or, upload a file:  No file selected. [Use a example sequence](#) | [Clear sequence](#) | [See more example inputs](#)

**AND**

Enter or paste your second nucleotide sequence in any supported format:

```
>Rosalind_37
ATGGACTCCTTGTGTCCTAAATAGATACATATTCACCAAGTGTGCCTTAGCCTTGGCGACTCCTTGTGTCCTAAATAGATACATATTTG
```

Or, upload a file:  No file selected.

**STEP 2 - Set your pairwise alignment options**

The default settings will fulfill the needs of most users.

(Click here, if you want to view or change the default settings.)

**STEP 3 - Submit your job**

☐ Be notified by email (Tick this box if you want to be notified by email when the results are available)

Εικόνα 19: Είσοδος 2 συμβολοσειρών DNA

## 1.12 Base Quality Distribution

Η ποιότητα των βάσεων μπορεί να ποικίλλει ανάλογα με τη θέση κατά την διάρκεια της ανάγνωσης εξαιτίας της φύσης της διαδικασίας αλληλουχίας. Δίνεται λοιπόν η δυνατότητα να ελέγξει κάποιος αυτήν την κατανομή ποιότητας χρησιμοποιώντας τη μονάδα "Per Base Sequence Quality" του προγράμματος FastQC. Οι μέσες αποδεκτές τιμές ποιότητας είναι 10 για το κατώτερο τεταρτημόριο και 25 για το μεσαίο.

Εάν οι τιμές πέσουν κάτω από αυτό το όριο, τότε η μονάδα επιστρέφει μια προειδοποίηση. Σημειώνεται πως για τις αναγνώσεις >50bp το FastQC θα ομαδοποιήσει τις βάσεις.

Δίνεται ως είσοδος ένα αρχείο FASTQ με όριο ποιότητας q και επιστρέφεται ο αριθμός θέσεων όπου η μέση ποιότητα βάσης πέφτει κάτω από το παραπάνω όριο.

### 1.13 Global Multiple Alignment

Σε αυτό το σημείο, χρησιμοποιείται το εργαλείο Clustal Omega για Multiple Sequence Alignment, του οποίου η λειτουργία φαίνεται παρακάτω.

Πιο συγκεκριμένα, γίνεται η επιλογή της λειτουργίας "Protein" ή "DNA" και, στη συνέχεια, η επικόλληση της ακολουθίας σε μια από τις αναφερόμενες μορφές (δίνεται και η δυνατότητα ανεβάσματος απευθείας του κατάλληλου αρχείου). Για την απόκτηση καλύτερης στοίχισης, κατάλληλη είναι η αργή επιλογή. Εάν το Clustal εκτελεστεί μόνο σε 2 ακολουθίες, τότε οι επιλογές παραμέτρων αντιστοιχούν με εκείνες του Needle (βλ. ενότητα 1.5).

Δίνεται ως είσοδος ένα σύνολο νουκλεοτιδικών ακολουθιών σε μορφή FASTA και επιστρέφεται το αναγνωριστικό της συμβολοσειράς που είναι πιο διαφορετικό από τις υπόλοιπες.

Χρησιμοποιείται η online διαθέσιμη ηλεκτρονική διεπαφή (εικόνα 20).

**Important note:** This tool can align up to 4000 sequences or a maximum file size of 4 MB.

**STEP 1 - Enter your input sequences**

Enter or paste a set of

PROTEIN

sequences in any supported format:

```
>Rosalind_18
GACATGTTTGTGCTTAACTCGTGGCGGCTAGCCGTAAGTAAAG
>Rosalind_23
ACTCATGTTTGTGCTTAACTCTTGGCGGCTAGCCGTAAGTAAAG
>Rosalind_51
TCCTATGTTTGTGCTTAACTCTTGGCGGCTAGCCGTAAGTAAAG
>Rosalind_7
CACGTCGTGTCGCTTAACTTTGATTGCGGCTAGCGTAGTTAGTTA
>Rosalind_28
GGGGTCATGCGCTTTGCTTAAACCTTGGCGGCTAGCCGTAATGTTT
```

Or, upload a file: [Browse...](#) No file selected. [Use a example sequence](#) | [Clear sequence](#) | [See more example inputs](#)

**STEP 2 - Set your parameters**

OUTPUT FORMAT

ClustalW with character counts

The default settings will fulfill the needs of most users.

[More options...](#) (Click here, if you want to view or change the default settings.)

**STEP 3 - Submit your job**

☐ Be notified by email (Tick this box if you want to be notified by email when the results are available)

[Submit](#)

Εικόνα 20: Είσοδος στο Clustal Omega

#### Results for job clustalo-l20220616-120316-0417-75001229-p1m

|  |  |                            |                                   |                                 |                                    |
|--|--|----------------------------|-----------------------------------|---------------------------------|------------------------------------|
| <a href="#">Alignments</a>                   | <a href="#">Result Summary</a>                       | <a href="#">Guide Tree</a> | <a href="#">Phylogenetic Tree</a> | <a href="#">Results Viewers</a> | <a href="#">Submission Details</a> |
| <a href="#">Download Alignment File</a>      | <a href="#">Show Colors</a>                          |                            |                                   |                                 |                                    |
| CLUSTAL 0(1.2.4) multiple sequence alignment |  |                            |                                   |                                 |                                    |
| Rosalind_7                                   | -----CACGTCGTGTCGCTTAACTTTGATTGCGGCTAGCGTAGTTAGTTA-- | 49                         |                                   |                                 |                                    |
| Rosalind_51                                  | --TCCTATGTTTGTGCTTAACTCT--TGGCGGCTAGCCGTA--AGGTAAG   | 49                         |                                   |                                 |                                    |
| Rosalind_23                                  | --ACTCATGTTTGTGCTTAACTCT--TGGCGGCTAGCCGTA--ACTTAAG   | 49                         |                                   |                                 |                                    |
| Rosalind_18                                  | ---GACATGTTTGTGCTTAACTCG--TGGCGGCTAGCCGTA--AGTTAAG   | 48                         |                                   |                                 |                                    |
| Rosalind_28                                  | GGGGTCATGCGCTTTGCTTAAACCT--TGGCGGCTAGCCGTA--ATGTTT-  | 50                         |                                   |                                 |                                    |

PLEASE NOTE: Showing colors on large alignments is slow.

Εικόνα 21: Αποτελέσματα

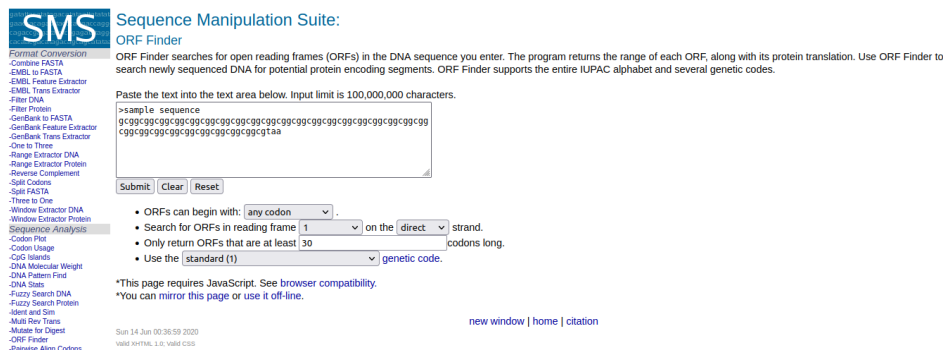
Όπως βλέπουμε, παίρνοντας τις ακολουθίες που δίνονται ως παραδείγματα από τη Rosalind και τοποθετώντας στο Clustal Omega, επιστρέφεται η πολλαπλή στοίχιση που προκύπτει, σύμφωνα με την είσοδο (εικόνα 21).

## 1.14 Finding Genes with ORFs

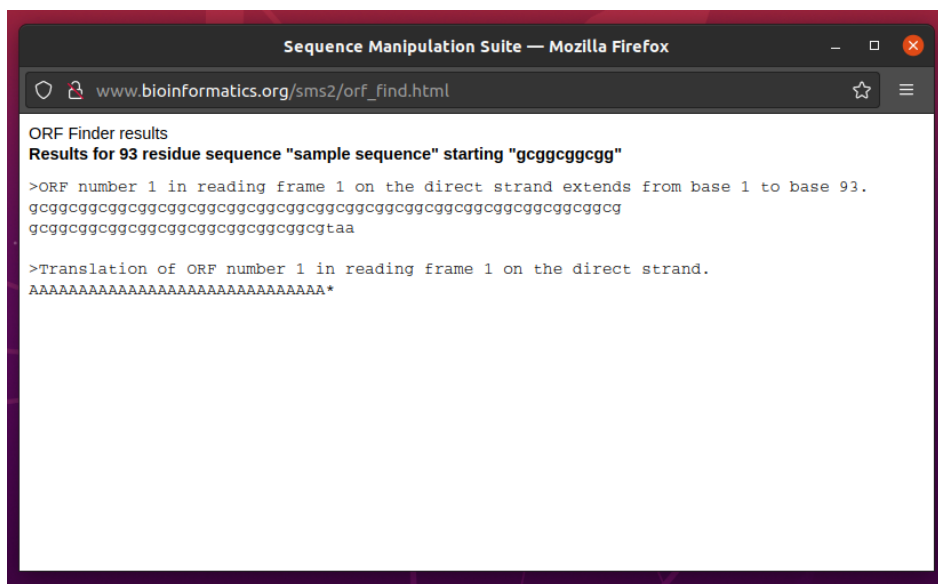
Το εργαλείο ORF finder από το πακέτο SMS 2 ανιχνεύει κωδικόνια από μία αλληλουχία βάσεων DNA, προκειμένου να προκύψει τελικά η πρωτεΐνη.

Δίνεται ως είσοδος μια συμβολοσειρά DNA s μήκους το πολύ 1 kbp και επιστρέφεται η μεγαλύτερη συμβολοσειρά πρωτεΐνης που μπορεί να μεταφραστεί από ένα ORF του s. Εάν υπάρχουν περισσότερες από μία πρωτεϊνικές συμβολοσειρές μέγιστου μήκους, τότε μπορείτε να εξαχθεί οποιαδήποτε λύση.

Χρησιμοποιείται η online διαθέσιμη ηλεκτρονική διεπαφή (εικόνα 22). Παρατηρούμε τελικά τα κωδικόνια που ανιχνεύτηκαν σύμφωνα με τα δεδομένα εισόδου (εικόνα 23).



Εικόνα 22: Είσοδος στο ORF finder



Εικόνα 23: Αποτελέσματα

## 1.15 Base Filtration by Quality

Σε αυτό το σημείο χρησιμοποιούμε το εργαλείο FASTQ Quality Trimmer, το οποίο "τρίμμάρει" μια ακολουθία σύμφωνα με παραμέτρους που αφορούν στην ποιότητα. Πιο συγκεκριμένα, οι βάσεις κακής ποιότητας μπορούν εύκολα να περικοπούν χρησιμοποιώντας συγκεκριμένο όριο (που ορίζεται από γραφική

παράσταση ποιότητας παρομοίας με αυτή που αναφερθήκαμε στην ενότητα 1.12). Υπάρχουν πολλά τέτοια εργαλεία, κάποια από αυτά ενδεικτικά είναι:

- Το FASTQ Quality Trimmer του Galaxy. Χρησιμοποιεί μια προσέγγιση "συρόμενου παραθύρου", επομένως για ένα απλό κόψιμο των άκρων θα πρέπει να οριστεί το μέγεθος του παραθύρου σε 1
- Το Trimmomatic. Είναι ένα εργαλείο που βασίζεται στη γραμμή εντολών σε Java. Για ένα απλό κόψιμο και από τα δύο άκρα θα πρέπει να καθοριστούν κατάλληλα οι παράμετροι LEADING και TRAILING.

Δίνεται ως είσοδος ένα αρχείο FASTQ με μια τιμή αποκοπής ποιότητας  $q$  και επιστρέφεται το ίδιο αρχείο κομμένο και από τα δύο άκρα (αφαιρέθηκαν οι leading και trailing βάσεις με ποιότητα χαμηλότερη από  $q$ ).

## 2 Άσκηση Δεύτερη

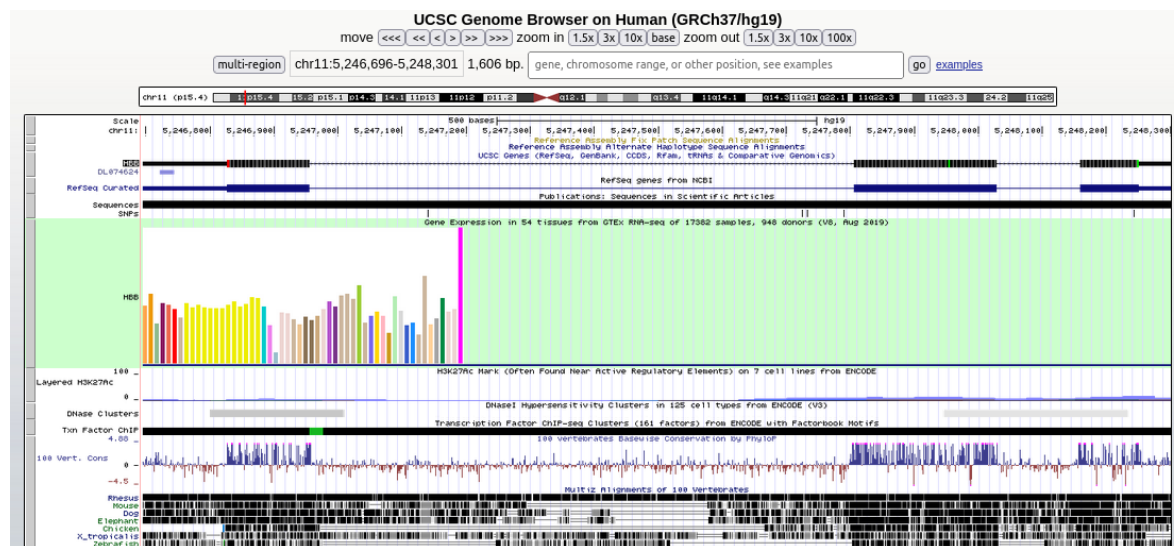
Σκοπός της συγκεκριμένης άσκησης είναι η στοίχιση ακολουθιών με χρήση του δικτυακού τόπου UCSC και του Genome Browser.

Το UCSC Genome Browser είναι ένα διαδικτυακό εργαλείο που χρησιμοποιείται σαν μικροσκόπιο πολλαπλής ισχύος, το οποίο επιτρέπει στους ερευνητές να προβάλλουν και τα 23 χρωμοσώματα του ανθρώπινου γονιδιώματος σε οποιαδήποτε κλίμακα από ένα πλήρες χρωμόσωμα έως ένα μεμονωμένο νουκλεοτίδιο.

Όσον αφορά στην άσκηση, θα γίνει επιλογή του human genome hg19 σύμφωνα με το ερώτημα hbb, το οποίο θα οδηγήσει στο αποτέλεσμα chr11:5,246,696–5,248,301. Το τελευταίο αντιπροσωπεύει μια περιοχή από 1606 ζευγών βάσεων που ορίζουν το beta globin γονίδιο. Στη συνέχεια, θα γίνει σύγκριση με διάφορους συνδυασμούς ειδών (όπως φαίνεται στις εικόνες 24 έως 28) σύμφωνα με τα Nets και Chains.

Τα Nets και Chains είναι συλλογές υψηλότερου επιπέδου βασικών ευθυγραμμίσεων ακολουθιών κατά ζεύγη. Τα Nets διασταυρούμενων ειδών χρησιμοποιούνται για τη δημιουργία μιας συλλογής μονής κάλυψης (single-coverage) - στο γονιδίωμα αναφοράς - από ζεύγη ευθυγραμμίσεων που αποτελούν τις βάσεις των ευθυγραμμίσεων πολλών ειδών Multiz στο κομμάτι της Διατήρησης. Οι αλγόριθμοι Nets και Chains, καθώς και τα αποτελέσματα από ευθυγραμμίσεις ανθρώπου - ποντικού δημιουργούνται από γονιδιωματικές τοπικές ευθυγραμμίσεις που υπολογίστηκαν από τον Blastz και τον Lastz, μετά την επεξεργασία από μια σειρά προγραμμάτων UCSC και κυρίως των axtChain, chainNet και netFilter.

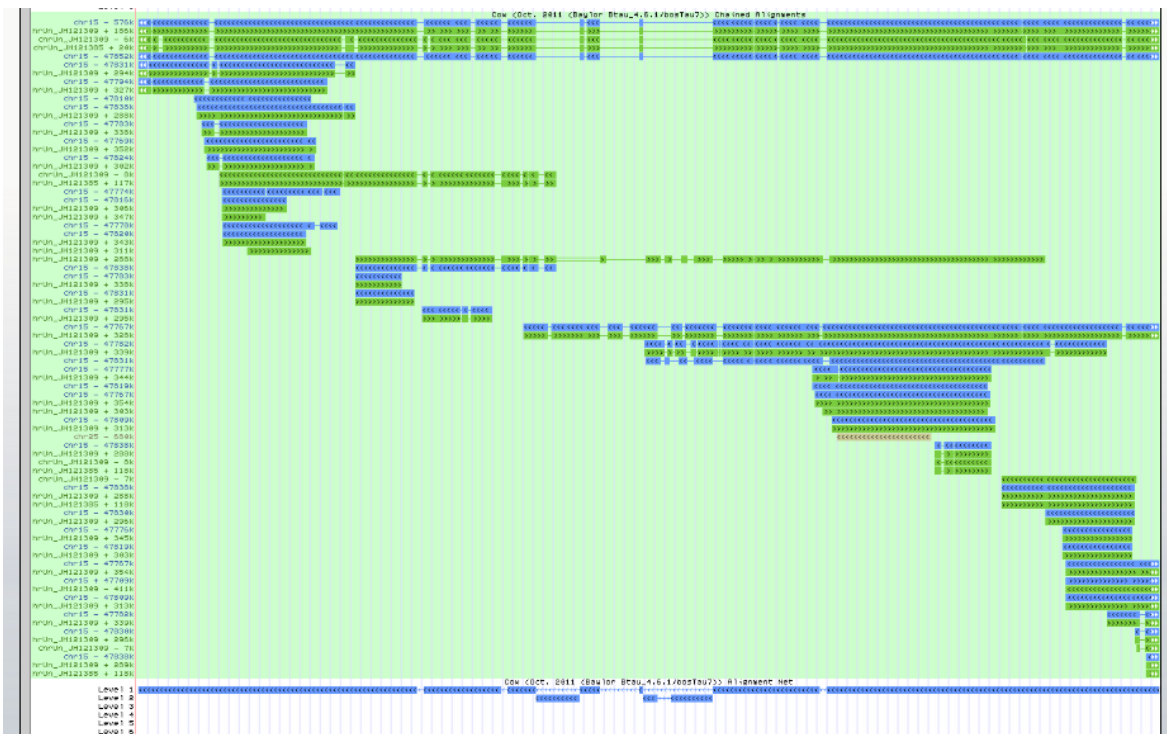
Στις παρακάτω εικόνες παρουσιάζονται τα αποτελέσματα στοίχισης της ακολουθίας ζευγών βάσεων που αποτελούν το γονίδιο HBB στον άνθρωπο, στο άλογο, στην αγελάδα, στο γουρούνι και στον αρουραίο. Όπως γίνεται προφανές υπάρχει μεγαλύτερη αντιστοιχία με την αγελάδα.



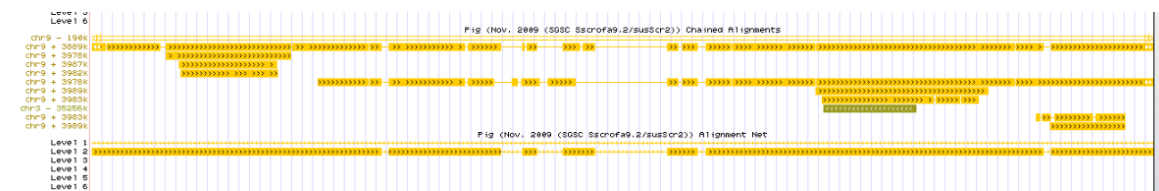
Εικόνα 24: Human genome hg19 με ερώτημα hbb



Εικόνα 25: Horse sample

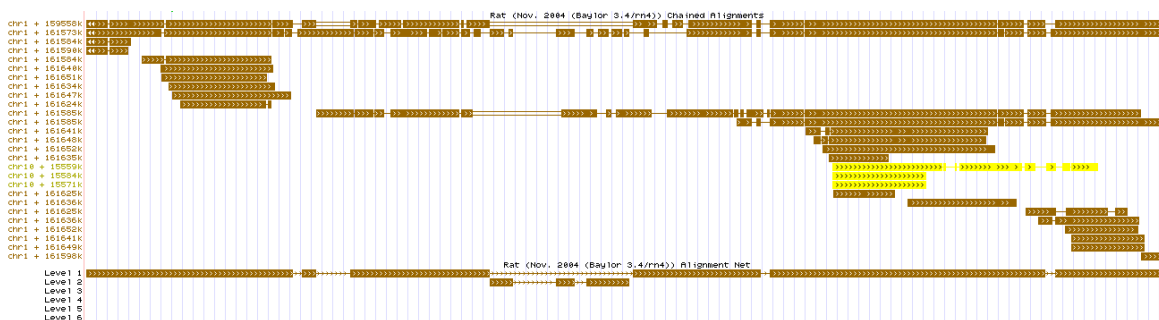


Εικόνα 26: Cow sample



Εικόνα 27: Pig sample





Εικόνα 28: Rat sample

### 3 Άσκηση Τρίτη

#### 3.1 Ερώτημα Α

Αρχικά, υλοποιείται το δέντρο επιθεμάτων (suffix tree) της συμβολοσειράς κειμένου  $T$  μήκους  $n$ , σύμφωνα με τον αλγόριθμο του Ukkonen. Στη συνέχεια, εφαρμόζεται ο αλγόριθμος DFS (Depth First Search - Αναζήτηση Κατά Βάθος) στο παραπάνω δέντρο, προκειμένου να ανατεθεί σε κάθε κόμβο μια ετικέτα (label), η οποία αντιπροσωπεύει τον ελάχιστο δείκτη επιθεμάτων που εμφανίζονται στους απογόνους του καθώς και την αντίστοιχη θέση του. Έπειτα, δεδομένης μιας συμβολοσειράς  $P$ , το δέντρο προσπελάσσεται έως ότου βρεθεί ο κόμβος που αντιστοιχεί στο  $P$  - ή κάποια προέκτασή του. Σε αυτό το σημείο λοιπόν ελέγχεται, αν η ετικέτα του συγκεκριμένου κόμβου υποδεικνύει εμφάνιση του  $P$  πριν από τη θέση  $k$ .

#### 3.2 Ερώτημα Β

Οι Ilie και Smyth<sup>2</sup> εισήγαγαν έναν αποδοτικό αλγόριθμο, ο οποίος εντοπίζει τη μικρότερη μη επαναλαμβανόμενη συμβολοσειρά σε ένα κείμενο βασιζόμενοι σε μια απλή παρατήρηση. Η παραπάνω παρατήρηση αναφέρει πως κάθε ελάχιστη μοναδική υποσυμβολοσειρά  $[i...j]$  είναι η μικρότερη μεταξύ όλων των μοναδικών επιθεμάτων (prefixes) που καταλήγουν στο  $j$  για όλα τα επιθέματα (suffixes) μιας λέξης  $w$ . Σύμφωνα με το παραπάνω λοιπόν εισήγαγαν έναν καινούργιο αλγόριθμο, στο τέλος του οποίου η μικρότερη μη-επαναλαμβανόμενη συμβολοσειρά  $[i...j]$  θα αποθηκευτεί, όταν  $\text{MinUnique}[j] = i$ .

Πιο συγκεκριμένα, στο βήμα 1 υπολογίζονται οι πίνακες SA (Suffix Array - περιέχει τις θέσεις 1, 2, ...,  $n$  ταξινομημένες σε αύξουσα λεξικογραφική σειρά των αντίστοιχων επιθεμάτων  $\text{suf}[i]$ ,  $i = 1, 2, \dots$ ) και LCP (Longest Common Prefix Array - περιέχει στη θέση  $i$  το μήκος του μακρύτερου κοινού προθέματος των  $\text{suf}[\text{SA}[i]]$  και  $\text{suf}[\text{SA}[i-1]]$ ). Έπειτα, στο βήμα 3 αρχικοποιείται ο πίνακας MINUNIQUE με τιμές μικρότερες από οποιεσδήποτε έγκυρες τιμές μπορεί να του ανατεθούν. Στο βήμα 5, υπολογίζεται το μεγαλύτερο επαναλαμβανόμενο πρόθεμα κάθε επιθέματος. Η μικρότερη μη-επαναλαμβανόμενη συμβολοσειρά βρίσκεται μια θέση μετά, εκτός της περίπτωσης που το μεγαλύτερο επαναλαμβανόμενο πρόθεμα φτάνει στο τέλος του  $w$ . Προκειμένου να αποφευχθεί ο παραπάνω έλεγχος σε κάθε βήμα, επιτρέπεται στο MINUNIQUE να έχει  $n+1$  στοιχεία, εκ των οποίων το τελευταίο θα αγνοείται. Τέλος, στο βήμα 6, ανανεώνεται ο πίνακας MINUNIQUE, αν βρεθεί μικρότερη συμβολοσειρά.

Κρίνεται απαραίτητο να σημειωθεί πως ο συγκεκριμένος αλγόριθμος έχει γραμμική χρονική πολυπλοκότητα.

<sup>2</sup>Ilie, Lucian & Smyth, Wf. (2011). Minimum Unique Substrings and Maximum Repeats



#### MINUNIQUE( $w$ )

1.     compute SA, LCP
2.     **for**  $i$  **from** 1 **to**  $n$  **do**
3.         MINUNIQUE[ $i$ ]  $\leftarrow$  0
4.     **for**  $i$  **from** 1 **to**  $n$  **do**
5.          $lcp \leftarrow \max(\text{LCP}[i], \text{LCP}[i + 1])$
6.         MINUNIQUE[SA[ $i$ ] +  $lcp$ ]  $\leftarrow \max(\text{MINUNIQUE}[\text{SA}[i] + lcp], \text{SA}[i])$
7.     **return** MINUNIQUE

Εικόνα 29: Αλγόριθμος MINUNIQUE

## 4 Άσκηση Τέταρτη

Οι Ho-Leung Chan, Wing-Kai Hon, Tak-Wah Lam και Kunihiko Sadakane <sup>3</sup> εισήγαγαν μια καινούργια έννοια, το Compressed Suffix Tree. Έστω  $C = T_1, T_2, \dots, T_k$  είναι μια συλλογή κειμένων με συνολικό μήκος  $n$ . Μπορούμε να διατηρήσουμε ένα συμπιεσμένο δέντρο επιθεμάτων για το  $C$ , το οποίο χρησιμοποιεί χώρο  $O(n)$ -bit και υποστηρίζει τα ακόλουθα ερωτήματα σχετικά με το δέντρο επιθεμάτων για το  $C$ : εύρεση της ρίζας σε χρόνο  $O(1)$ , και εύρεση του γονέα, του αριστερού παιδιού, του αριστερού αδελφού, του δεξιού αδελφού και του επιθέματος συνδέσμου ενός κόμβου σε χρόνο  $O(\log n)$ . Η ετικέτα ακμής και το φύλλο μπορούν να υπολογιστούν σε χρόνο  $O(\log_2 n)$ . Η εισαγωγή ή διαγραφή ενός κειμένου  $T$  στο  $C$  μπορεί να γίνει σε  $O(|T| \log_2 n)$  χρόνο.

Ακόμα, η δενδροειδής δομή ενός suffix tree αναπαριστάται από μια λίστα παρενθέσεων ως εξής: διασχίζοντας το suffix tree με τον αλγόριθμο DFS, την πρώτη φορά που επισκέπτεσαι έναν κόμβο, προσθέτεις ένα "(" στη λίστα και στη τελευταία φορά που επισκέπτεσαι έναν, προσθέτεις ένα ")" (αντίστοιχα στην λίστα. Σημειώνεται πως η λίστα των παρενθέσεων είναι ισορροπημένη και κάθε κόμβος στο δένδρο επιθεμάτων αντιπροσωπεύεται από ένα ζεύγος ταιριασμένων παρενθέσεων. Επομένως, μπορούμε να καθορίσουμε έναν κόμβο  $u$  στο δέντρο επιθεμάτων χρησιμοποιώντας τη θέση ανοικτής παρένθεσης που αντιπροσωπεύει το  $u$ .

Η παραπάνω λίστα των ισορροπημένων παρενθέσεων υποστηρίζει διάφορα ερωτήματα σχετικά με το suffix tree. Συγκεκριμένα, ο χαμηλότερος κοινός πρόγονος δύο κόμβων  $u$  και  $v$  είναι το double-enclose( $u, v$ ). Η τάξη ενός φύλλου  $u$ , η οποία είναι η λεξικογραφική σειρά του επιθέματος που αντιστοιχεί σε αυτό, είναι rank-leaf( $u$ ). Το  $i$ -οστό φύλλο, το οποίο είναι αυτό που αντιστοιχεί στο λεξικογραφικά  $i$ -οστό επίθεμα, δίνεται από το select-leaf( $i$ ). Το αριστερότερο φύλλο και το δεξιότερο φύλλο του υπό-δένδρου με ρίζα στο  $u$  μπορούν να βρεθούν από τις σχέσεις rank-leaf( $u-1$ )+1 και rank-leaf(find-match( $u$ ))), αντίστοιχα. Κάθε μία από τις παραπάνω λειτουργίες απαιτεί  $O(\log n)$  χρόνο.

Όσον αφορά στις πράξεις που μπορούν να γίνουν ισχύουν τα παρακάτω. Ας υποθέσουμε ότι έχουμε τη λίστα των ισορροπημένων παρενθέσεων, CSA (Compressed Suffix Array), FM-index <sup>4</sup> και LCP <sup>5</sup> που αναπαριστούν το suffix tree για μια συλλογή κειμένων  $C$ . Για να εισάγουμε ένα νέο κείμενο  $T$  στο  $C$ , ενημερώνουμε τις δομές δεδομένων ώστε να αντικατοπτρίζουν την αλλαγή ότι όλα τα επιθέματα του  $T$  εισάγονται στο suffix tree. Στη συνέχεια, εκτελούμε την ενημέρωση σε γύρους  $|T|$  έτσι ώστε στον  $i$ -οστό γύρο, το  $i$ -οστό συντομότερο επίθεμα  $T[|T| - i + 1 .. |T|]$  εισάγεται ως νέο φύλλο στο suffix tree. Κάθε γύρος περιλαμβάνει την ενημέρωση του καταλόγου των ισορροπημένων παρενθέσεων, CSA, FM-index και LCP. Έτσι, στο τέλος του  $i$ -οστού γύρου, οι δομές δεδομένων αναπαριστούν το συμπιεσμένο δένδρο επιθέτων για τη συλλογή  $C \cup T[|T| - i + 1 .. |T|]$ . Το βασικό μέλημα είναι η ενημέρωση του καταλόγου των ισορροπημένων παρενθέσεων και του LCP, η οποία γίνεται με τα ακόλουθα δύο βήματα: υπολογισμός των νέων πληροφοριών του suffix tree και την ενημέρωση των δομών δεδομένων σύμφωνα με το νέο suffix tree. Για το πρώτο βήμα, παρατηρούμε ότι το συμπιεσμένο δέντρο επιθεμάτων υποστηρίζει

<sup>3</sup>Dynamic Dictionary Matching and Compressed Suffix Trees

<sup>4</sup>CSA & FM-index: Αυτοί οι δείκτες είναι συμπιεσμένες εκδόσεις πινάκων επιθεμάτων που καταλαμβάνουν μόνο  $O(n)$  bits, αλλά υποστηρίζουν αποτελεσματική αναζήτηση προτύπων

<sup>5</sup>διατηρεί το μήκος του μεγαλύτερο κοινού προθέματος μεταξύ δύο γειτονικών φύλλων

τις λειτουργίες πλοήγησης στο κανονικό δέντρο επιθέτων, οπότε μπορούμε να χρησιμοποιήσουμε τον αλγόριθμο του Weiner για να υπολογίσουμε τη θέση του νέου φύλλου.

## 5 Άσκηση Πέμπτη

Για την επίλυση του συγκεκριμένου ερωτήματος έχει δημιουργηθεί ο παρακάτω αλγόριθμος, ο οποίος ακολουθεί τα εξής βήματα:

- Δημιουργία ενός γενικευμένου δέντρου επιθεμάτων σύμφωνα με τον αλγόριθμο του Ukkonen (GST - Generalized Suff Tree) για μία συμβολοσειρά  $S$  και ένα πρότυπο  $P$ . Σε κάθε εσωτερικό κόμβο  $v$  του GST με 1 (2), σημειώνεται όταν υπάρχει φύλλο στο αντίστοιχο υπό-δέντρο του  $v$ , το οποίο αναπαριστά ένα suffix του  $S$  ( $P$ )
- Πέρασμα του GST από το πιο απομακρυσμένο φύλλο προς τα πάνω ψάχνοντας την αντίθετη συμβολοσειρά από αυτή που επιθυμούμε (π.χ. αν αναζητώ το πρότυπο  $P = \text{aqra}$  τότε θα αναζητήσω την συμβολοσειρά  $\text{arqa}$ )
- Σύγκριση κάθε φορά του υπό-μονοπατιού που βρισκόμαστε, προκειμένου να ταιριάζει με το πρότυπο που επιθυμούμε
- Αποθήκευση της θέσης/εσωτερικού κόμβου του υπό-μονοπατιού, στο οποίο εντοπίζεται το πρότυπο

Από τη στιγμή λοιπόν που τρέχω το δέντρο από κάτω προς τα πάνω το μονοπάτι, του οποίου το υπό-μονοπάτι του περιέχει το πρότυπο  $P$ , θα διέρχεται από τη ρίζα.

## 6 Άσκηση Έκτη

### 6.1 Ερώτημα Α

Το απλό πρόβλημα της καθολικής στοίχισης χωρίς συγγενική ποινή ασυμφωνίας θα λυθεί με χρήση τεχνικών δυναμικού προγραμματισμού.

Σε πρώτο επίπεδο εφαρμόζουμε ουσιαστικά τον αλγόριθμο των Needleman-Wunsch ως εξής. Κατασκευάζεται ένας γράφος οι κόμβοι του οποίου αντιπροσωπεύουν τα ζεύγη χαρακτήρων των δύο ακολουθιών. Καθένας από τους κόμβους διαθέτει ακμές που οδηγούν στον κάτω, κάτω δεξιά και δεξιά κόμβο του γράφου. Οι παραπάνω ακμές θα χαρακτηρίζονται από κάποια βάρη, τα οποία υπολογίζονται σύμφωνα με την πράξη που θα πραγματοποιείται κάθε φορά (αντικατάσταση ενός χαρακτήρα, προσθήκη κενού και αφαίρεση χαρακτήρα). Πρακτικά, λοιπόν, δημιουργείται ένας γράφος, από τον οποίο, ξεκινάμε από τον κάτω δεξιά κόμβο του και ακολουθούμε διαγώνια διαδρομή, όσο οι χαρακτήρες δεν διαφέρουν. Σε περίπτωση που υπάρξει ασυμφωνία, γίνεται επιλογή του ελάχιστου κόστους (αριστερά, πάνω ή πάνω αριστερά κόμβος) και εκτελείται η αντίστοιχη πράξη (προσθήκη κενού όσον αφορά στις δύο πρώτες περιπτώσεις ή ασυμφωνία χαρακτήρα στην στην τελευταία περίπτωση). Επιπλέον, πρέπει να αναφερθεί πως ο συγκεκριμένος αλγόριθμος υλοποιείται σε  $O(nm)$ , όπου  $n, m$  τα μήκη των ακολουθιών.

Στη συγκεκριμένη περίπτωση όμως ο παραπάνω αλγόριθμος πρέπει να επεκταθεί, μιας και στον γράφο πρέπει να εισαχθούν νέες ακμές που θα ενώνουν διαγώνια κόμβους με μεγαλύτερες αποστάσεις. Γι' αυτό το λόγο, θα εφαρμόσουμε μια μέθοδο σαν κι αυτή των affine gap penalties. Αυτό σημαίνει πως θα δημιουργήσουμε έναν δεύτερο γράφο, σύμφωνα με τον οποίο θα αποθηκεύουμε τη πληροφορία των ασυμφωνιών. Κατά αυτό τον τρόπο, ορίζουμε τους παρακάτω κανόνες.

Σε περίπτωση που υπάρξει ανάγκη για προσθαφαίρεση, το κόστος ισούται με:

$$S_{i,j} = S_{i-1,j} - \rho$$

ή

$$S_{i,j} = S_{i,j-1} - \rho$$

Αν υπάρχει συμφωνία χαρακτήρων, το κόστος γίνεται:

$$S_{i,j} = S_{i-1,j-1} + 1$$

Όπως αναφέραμε και παραπάνω, σε περίπτωση ασυμφωνίας οδηγούμαστε στον δεύτερο γράφο και το κόστος γίνεται:

$$W_{i,j} = S_{i,j} - \rho$$

Σημειώνεται πως ο αλγόριθμος απαιτεί  $O(nm)$  για την κατασκευή των δύο γράφων, συνεπώς αυτή είναι και η τελική του πολυπλοκότητα.

## 6.2 Ερώτημα B

Δημιουργήσαμε αλγόριθμο, ο οποίος υπολογίζει το μέγιστο κοινό πρόθεμα από ένα σύνολο συμβολοσειρών. Πιο συγκεκριμένα, χρησιμοποιήσαμε τη μέθοδο Divide and Conquer, μιας και η ιδέα είναι να διαιρέσουμε όλες τις συμβολοσειρές σε δύο μικρότερα σύνολα και στη συνέχεια να επεξεργαστούμε αναδρομικά αυτά τα σύνολα. Παρακάτω, παρουσιάζεται ο κώδικας:

```
# All possible pairs in List
# Using list comprehension + enumerate()
def getEachPair(list):

    for index, X in enumerate(list):
        for Y in list[index+1:]:
            res = LCP(X, Y)
            print("The longest common prefix between", X, "and", Y, "is", res)

# function to find the longest common prefix (LCP)
# between two strings
def LCP(X, Y):

    k = 1
    l = 0

    while k < len(X) and l < len(Y):
        if X[k] != Y[l]:
            break
        k = k + 1
        l = l + 1

    return X[:k]
```

## 7 Άσκηση Επτά

Ο Eugene W. Myers<sup>6</sup> προσπάθησε να αποδείξει ότι  $D(n,m)=m+n-2u$  ή  $u=(m+n-D(n,m))/2$  βασιζόμενος στα εξής. Ένα edit script για δύο συμβολοσειρές A και B είναι ένα σύνολο εντολών και διαγραφής που μετατρέπουν το A σε B. Η διαγραφή "xD" (deletion) διαγράφει το σύμβολο  $a_x$  από το A. Η εντολή εισαγωγής " $xIb_1, b_2, \dots, b_t$ " (insertion) εισάγει την ακολουθία συμβόλων  $b_1, \dots, b_t$  αμέσως μετά το  $a_x$ . Κάθε ίχνος<sup>7</sup> αντιστοιχεί μοναδικά σε ένα edit script. Έστω  $(x_1, y_1)(x_2, y_2)\dots(x_L, y_L)$  είναι ίχνος με  $y_0 = 0$  και  $y_L + 1 = M + 1$ . Το script αποτελείται από τις εντολές "xD" για  $x \notin x_1, x_2, \dots, x_L$ , και " $x_kIb_{y_k+1}, \dots, b_{y_k+1}$ " για k τέτοιο ώστε  $y_k + 1 < y_{k+1}$ . Το script διαγράφει σύμβολα N-L και εισάγει σύμβολα M-L. Συνεπώς, για κάθε ίχνος μήκους L υπάρχει ένα αντίστοιχο script μήκους  $D = N + M - 2L$ .

Η μέγιστη κοινή υπό-συμβολοσειρά μεταξύ δύο συμβολοσειρών  $S_1, S_2$  μπορεί να βρεθεί σε γραμμικό χρόνο χρησιμοποιώντας ένα GST (generalized suffix tree). Πιο συγκεκριμένα,

<sup>6</sup>Eugene W. Myers, An  $O(ND)$  Difference Algorithm and Its Variations

<sup>7</sup>Ένα ίχνος μήκους L είναι μια ακολουθία σημείων αντιστοίχισης  $L, (x_1, y_1)(x_2, y_2)\dots(x_L, y_L)$ , έτσι ώστε  $x_i < x_{i+1}$  και  $y_i < y_{i+1}$  για διαδοχικά σημεία  $(x_i, y_i)$  και  $(x_{i+1}, y_{i+1})$  για  $i \in [1, L-1]$

## 1. Κατασκευή ενός Generalized Suffix Tree

2. Σημείωσε σε κάθε εσωτερικό κόμβο  $v$  του GST με 1 (2), όταν υπάρχει φύλλο στο αντίστοιχο υπό-δέντρο του  $v$ , το οποίο αναπαριστά ένα suffix του  $S_1$  ( $S_2$ ).

Κάθε εσωτερικός κόμβος που σημειώνεται ταυτόχρονα με 1 και 2, δηλώνει την ύπαρξη ενός κοινού substring μεταξύ των συμβολοσειρών  $S_1$  και  $S_2$ . Ο κώδικας που υλοποιεί το ζητούμενο περιλαμβάνεται στο ίδιο Archive με αυτό το pdf και όνομα suffix\_trees.

## 8 Άσκηση Οκτώ

Η ακολουθία της spike (ακίδα) πρωτεΐνης του κορονοϊού έχει ληφθεί από [εδώ](#) και βρίσκεται στο φάκελο MerosA\_Askisi8 με όνομα spike ενώ αντίστοιχα για τον συναφές κορονοϊό Bat-RaTG13 έχει ληφθεί από [εδώ](#) και βρίσκεται στο φάκελο MerosA\_Askisi8 με όνομα bat. Στον φάκελο υπάρχει το αρχείο main στο οποίο εκτελείται ο ζητούμενος κώδικας και προκύπτουν τα αντίστοιχα αποτελέσματα.

## 9 Άσκηση Εννέα

### 9.1 Ερώτημα Α

Εφαρμόζουμε τον αλγόριθμο Needleman-Wunsch, για τον οποίο ισχύουν τα εξής:

$$D_{i,j} = \max \begin{cases} D_{i-1,j-1} + s(v_i, w_j) \\ D_{i-1,j} + s(v_i, -) \\ D_{i,j-1} + s(-, w_j) \end{cases} = \max \begin{cases} D_{i-1,j-1} + 1, v_i = w_j \\ D_{i-1,j-1} - 1, v_i \neq w_j \\ D_{i-1,j} - 1, w_i = - \\ D_{i,j-1} - 1, v_i = - \end{cases}$$

Σύμφωνα με τα παραπάνω προκύπτει ο εξής πίνακας:

| $D_{i,j}$ |     | G        | A        | T        | C        | G        | T        | G        | A        | A        | T        | T        |
|-----------|-----|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
|           | 0   | -1       | -2       | -3       | -4       | -5       | -6       | -7       | -8       | -9       | -10      | -11      |
| G         | -1  | <b>1</b> | 0        | -1       | -2       | -3       | -4       | -5       | -6       | -7       | -8       | -9       |
| G         | -2  | 0        | <b>0</b> | -1       | -2       | -1       | -2       | -3       | -4       | -5       | -6       | -7       |
| T         | -3  | -1       | -1       | <b>1</b> | 0        | -1       | 0        | -1       | -2       | -3       | -4       | -5       |
| T         | -4  | -2       | -2       | <b>0</b> | 0        | -1       | 0        | -1       | -2       | -3       | -2       | -3       |
| C         | -5  | -3       | -3       | -1       | <b>1</b> | 0        | -1       | -1       | -2       | -3       | -3       | -3       |
| G         | -6  | -4       | -4       | -2       | 0        | <b>2</b> | 1        | 0        | -1       | -2       | -3       | -4       |
| T         | -7  | -5       | -5       | -3       | -1       | 1        | <b>3</b> | 2        | 1        | 0        | -1       | -2       |
| G         | -8  | -6       | -6       | -4       | -2       | 0        | 2        | <b>4</b> | 3        | 2        | 1        | 0        |
| G         | -9  | -7       | -7       | -5       | -3       | -1       | 1        | <b>3</b> | 3        | 2        | 1        | 0        |
| A         | -10 | -8       | -6       | -6       | -4       | -2       | 0        | 2        | <b>4</b> | <b>4</b> | <b>3</b> | <b>2</b> |

Score = 2

G A T - C G T G - A - - -

G G T T C G T G G A A T T

## 9.2 Ερώτημα Β

Εφαρμόζουμε τον αλγόριθμο Smith - Waterman, για τον οποίο ισχύουν τα εξής:

$$S_{i,j} = \max \begin{cases} S_{i-1,j-1} + s(v_i, w_j) \\ S_{i-1,j} + s(v_i, -) \\ S_{i,j-1} + s(-, w_j) \\ 0 \end{cases} = \max \begin{cases} S_{i-1,j-1} + 1, v_i = w_j \\ S_{i-1,j-1} - 1, v_i \neq w_j \\ S_{i-1,j} - 1, w_i = - \\ S_{i,j-1} - 1, v_i = - \\ 0 \end{cases}$$

Σύμφωνα με τα παραπάνω προκύπτει ο εξής πίνακας:

| <u>D<sub>i,j</sub></u> |   | G | A | T | C | G | T | G | A | A | T | T |
|------------------------|---|---|---|---|---|---|---|---|---|---|---|---|
|                        | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| G                      | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| G                      | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| T                      | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 1 | 0 | 0 | 1 | 1 |
| T                      | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 2 |
| C                      | 0 | 0 | 0 | 0 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| G                      | 0 | 1 | 0 | 0 | 1 | 3 | 2 | 1 | 0 | 0 | 0 | 0 |
| T                      | 0 | 0 | 0 | 1 | 0 | 2 | 4 | 3 | 2 | 1 | 1 | 1 |
| G                      | 0 | 1 | 0 | 0 | 0 | 1 | 3 | 5 | 4 | 3 | 2 | 1 |
| G                      | 0 | 1 | 0 | 0 | 0 | 1 | 2 | 4 | 4 | 3 | 2 | 1 |
| A                      | 0 | 0 | 2 | 1 | 0 | 0 | 1 | 3 | 5 | 5 | 4 | 3 |

Score = 5

C G T G - A -

C G T G G A A