

1η Εργασία βιοπληροφορικής

Άσκηση 1

Το πρώτο πρόβλημα είναι δεδομένης μίας αλληλουχίας DNA να μετρήσουμε τις εμφανίσεις του κάθε αμινοξέος.

Ας δούμε πρώτα με χρήση biopython

```
1 from Bio.Seq import Seq
2
3 my_seq = Seq("AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAGAGTGTCTGATAGCAGC")
4
5 print("Adenine :" + str(my_seq.count("A")))
6 print("Cytosine :" + str(my_seq.count("C")))
7 print("Guanine :" + str(my_seq.count("G")))
8 print("Thymine :" + str(my_seq.count("T")))
```

Adenine :20
Cytosine :12
Guanine :17
Thymine :21

Το Seq είναι μια δομή που βολεύει όταν διαχειριζόμαστε αλληλουχίες διότι μας επιτρέπει να χρησιμοποιήσουμε έτοιμες συναρτήσεις. Η str() είναι απαραίτητη διότι έχουμε δύο διαφορετικού τύπου αντικείμενα.

Θα δοκιμάσουμε και το Sequence Manipulation Suite ένα γνωστό εργαλείο για ανάλυση αλληλουχιών. Θα χρησιμοποιήσουμε το DNA stats που είναι online.

SMS

Format Conversion

- Combine FASTA
- EMBL to FASTA
- EMBL Feature Extractor
- EMBL Trans Extractor
- Filter DNA
- Filter Protein
- GenBank to FASTA
- GenBank Feature Extractor
- GenBank Trans Extractor
- One to Three
- Range Extractor DNA
- Range Extractor Protein
- Reverse Complement
- Split Codons
- Split FASTA
- Three to One
- Window Extractor DNA
- Window Extractor Protein

Sequence Analysis

- Codon Plot
- Codon Usage
- CpG Islands
- DNA Molecular Weight
- DNA Pattern Find
- DNA Stats
- Fuzzy Search DNA
- Fuzzy Search Protein
- Ident and Sim
- Multi Rev Trans
- Mutate for Digest
- ORF Finder
- Pairwise Align Codons
- Pairwise Align DNA
- Pairwise Align Protein
- PCR Primer Stats
- PCR Products
- Protein GRAVY
- Protein Isoelectric Point
- Protein Molecular Weight
- Protein Pattern Find
- Protein Stats
- Restriction Digest
- Restriction Summary
- Reverse Translate
- Translate

Sequence Figures

- Color Align Conservation
- Color Align Properties
- Group DNA
- Group Protein
- Primer Map
- Restriction Map
- Translation Map

Random Sequences

- Mutate DNA
- Mutate Protein
- Random Coding DNA
- Random DNA Sequence
- Random DNA Regions
- Random Protein Sequence
- Random Protein Regions

Sequence Manipulation Suite:
DNA Stats

DNA Stats returns the number of occurrences of each residue in the sequence you enter. Percentage totals are also given for each

Paste the raw sequence or one or more FASTA sequences into the text area below. Input limit is 500,000,000 characters.

>sample sequence
AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAGAGTGTCTGA
TAGCAGC

Submit Clear Reset

*This page requires JavaScript. See browser compatibility.
*You can mirror this page or use it off-line.

Sequence Manipulation Suite - Avast Secure Browser

about:blank

DNA Stats results
Results for 70 residue sequence "sample sequence" starting "AGCTTTTCAT"

Pattern:	Times found:	Percentage:
g	17	24.29
a	20	28.57
t	21	30.00
c	12	17.14
n	0	0.00
u	0	0.00
r	0	0.00
y	0	0.00
s	0	0.00
w	0	0.00
k	0	0.00
m	0	0.00
b	0	0.00
d	0	0.00
h	0	0.00

Όπως βλέπουμε αυτό το εργαλείο μας δείχνει και τα ποσοστά ύπαρξης του κάθε αμινοξέος.

Έπειτα θα αναζητήσουμε όλες τις καταχωρίσεις ενός γένους που έγιναν μεταξύ 2 ημερομηνιών, στην GenBank του NCBI.

Ψάχνουμε στην online σελίδα.

NIH

National Center for Biotechnology Information

Nucleotide

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

Create alert Advanced

Species

- Plants (7)
- Customize ...

Molecule types

- genomic DNA/RNA (7)
- Customize ...

Source databases

- INSDC (GenBank) (7)
- Customize ...

Sequence Type

- Nucleotide (7)

Genetic compartments

- Chloroplast (7)
- Plastid (7)

Sequence length

- Custom range...

Release date

- Custom range...

Revision date

- Custom range...

[Clear all](#)

Summary 20 per page Sort by Default order

Items: 7

- ☐ [Anthoxanthum odoratum Rec A1 chloroplast microsatellite sequence](#)
1. 113 bp linear DNA
Accession: AY243051.1 GI: 33413983
[Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum 1973 Aa chloroplast microsatellite sequence](#)
2. 107 bp linear DNA
Accession: AY243050.1 GI: 33413982
[Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum 1925 A1 chloroplast microsatellite sequence](#)
3. 105 bp linear DNA
Accession: AY243049.1 GI: 33413981
[Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)
- ☐ [Anthoxanthum odoratum Rc16 1900 A2/1925 A2 chloroplast microsatellite sequence](#)
4. 113 bp linear DNA
Accession: AY243048.1 GI: 33413980

Send to: Filters: [Manage Filters](#)

Results by taxon

- Top Organisms [\[Tree\]](#)
 - Anthoxanthum odoratum (6)
 - Anthoxanthum nitens (1)

Analyze these sequences

- Run BLAST

Find related data

- Database: [Select](#)
- [Find items](#)

Search details

- "Anthoxanthum"[Organism] AND ("2003/7/25"[PDAT] : "2005/12/27"[PDAT])
- [Search](#)

Όπως βλέπουμε βάλαμε φίλτρο για αναζήτηση στην GenBank, με το “nucleotide”.

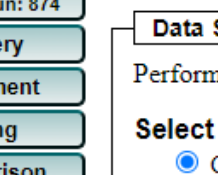
Η συνάρτηση Bio.Entrez.esearch() της biopython μπορεί να ψάξει όλες τις βάσεις δεδομένων του NCBI.

Το επόμενο πρόβλημα είναι να δώσουμε στην genBank 3 id και να μας επιστραφεί η μικρότερη αλληλουχία σε FASTA format. Θα ξεκινήσουμε με την biopython. Στην αρχή δοκιμάσαμε τον παρακάτω κώδικα ο οποίος δεν χρησιμοποιεί όλα τα εργαλεία που προτείνονται στο Rosalind. Σύντομα παρατηρήσαμε ότι παρόλο που δίνει σωστό αποτέλεσμα αυτός ο κώδικας μετράει και το μέγεθος της περιγραφής μαζί με αυτό της αλληλουχίας.

Άρα αλλάζουμε τον κώδικα σε:

Και πλέον ο κώδικας είναι σωστός διότι μετράει μόνο το μήκος της αλληλουχίας.

Πρώτα φτιάχνουμε ένα fasta αρχείο με τις ακολουθίες μας.



MEME
Multiple Em for Motif Elicitation

MEME discovers novel, **ungapped** motifs (recurring, fixed-length patterns) in your sequences (sample output from sequences). MEME splits variable-length patterns into two or more separate motifs. See this [Manual](#) for more information.

MEME Suite 5.5.5

Jobs running: 14
Jobs waiting to run: 874

- Motif Discovery
- Motif Enrichment
- Motif Scanning
- Motif Comparison
- Gene Regulation
- Utilities
- Manual
- Guides & Tutorials
- Sample Outputs
- File Format Reference
- Databases
- Download & Install
- Help
- Alternate Servers
- Authors & Citing
- ▼ Recent Jobs
- Clear All
- ↩ Previous version 5.5.4

Data Submission Form

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
 Επιλογή αρχείου
Δεν επιλέχθηκε κανένα αρχείο.

Input the primary sequences

Enter sequences in which you want to find motifs. ?

Upload sequences ▼
Επιλογή αρχείου
gg.fasta
PROTEIN ?

Select the site distribution

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

Zero or One Occurrence Per Sequence (zoops) ▼

Select the number of motifs

How many motifs should MEME find? ?

Input job details

(Optional) Enter your email address. ?

(Optional) Enter a job description. ?

► **Advanced options**

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

Start Search
Clear Input

Version 5.5.5
Please send comments and questions to: meme-suite@uw.edu
Powered by Opal

Στην επόμενη άσκηση ψάχνουμε να βρούμε αν 2 αλληλουχίες έχουν κοινούς προγόνους. Έτσι θέλουμε να κάνουμε ευθυγράμμιση και να δούμε τι σκορ λαμβάνουν.

Θα χρησιμοποιήσουμε τα εργαλεία Needle και Stretcher. Μια διαφορά τους είναι ότι το Stretcher χρησιμοποιεί διαφορετική βαθμολόγηση όταν τελειώνουν τα κενά ενώ το Needle δεν το έχει ως

προεπιλογή. Επίσης στις επιλογές που μας δίνουν τα δύο εργαλεία έχουν διαφορετικά νούμερα. Πχ το Needle έχει στην αναζήτησή μας GAP EXTEND=0.5 ενώ το Stretcher έχει επιλογές 1,2,3 κτλ.

↵🔒https://www.ebi.ac.uk/jdispatcher/psa/emboss_needle🔒🌐🌟🔗

CTCCCTCCCGACTTCCTTTGCCTTTCGTCGCCGTCCAGTCCACCTTCTCGTCCAACCTCTCGTCAAAC
TCCTCCAGCGCCTACACCAACACGGCAGGAAGAGCCGGCGCGAGCCCTCCGAGCCTGCTTCGGCCGGAG
AAGGGTTTGATGCGCTCGATGACATCGACCAGCTCCTCGACTTCGCGTCGCTTTCCATGCCGTGGGACTC
CGAGCCGTTCCCGGGGGTTAGCATGATGCTAGAGAACGCCATGTCGGCGCCGCCGACGCCGTGGGCGAC

Επιλογή αρχείουΔεν επιλέχθηκε κανένα αρχείο.

Use the exampleClear sequenceMore examp

Parameters

OUTPUT FORMAT ⓘMATRIX ⓘGAP OPEN ⓘGAP EXTEND ⓘEND GAP ⓘ

pair▼DNAfull▼10▼0.5▼false▼

END GAP OPEN ⓘEND GAP EXTEND ⓘ

10▼0.5▼

Less options ⤴

Submit

Title

EMBOSS Needle's job

Submit

→🔄🔒https://www.ebi.ac.uk/jdispatcher/psa/emboss_stretcher🔒🌐🌟🔗

TAATCAAACTCTATGTTTAGTTTTGCATGTAAAAAAAAAAAAAAAAAAAAAAAAA

Επιλογή αρχείουΔεν επιλέχθηκε κανένα αρχείο.

Paste your sequence here - or use the example sequence

AGCGGAGGTTACCTGCCGGAGCTGAAGACGAGGGATGGCATCTCCATCCCATGGAGGACATCGGAACG
TCGCGCGTGTGGAACATGCGGTACAGGTTTTGGCCCAACAACAAGAGCAGAATGTATCTGCTGGAGAACA
CAGGGGAATTTGTTCTTCCAACGAGCTTCAGGAGGGGGATTTCATAGTGATCTACTCCGATGTCAAGTC
GGGCAATATCTGATACGGGGCGTGAAGGTAAGGCCCCCGCGCGCAAGAGCAAGGCAGTGGTTCCAGC
GGGGGAGGCAAGCACAGGCCCTCTGTCCAGCAGGTCCAGGGAGAGCCGACGCCGCGGTGCTCCTGAAG
ACGCCGTGCTGCACGGGGTCAGCGGCGCCTGCAAGGGGAGGTCTCCGGAAGCGTGCGGCGGGTTCGGCA
GCAGGGAGCCGGCGCCATGAGCCAGATGGCGGTGAGCATC

Επιλογή αρχείουΔεν επιλέχθηκε κανένα αρχείο.

Use the exampleClear sequenceMore

Parameters

OUTPUT FORMAT ⓘMATRIX ⓘGAP OPEN ⓘGAP EXTEND ⓘ

pair▼DNAfull▼16▼4▼

Less options ⤴

Submit

Title

EMBOSS Stretcher's job

Submit

Ξεκινάμε με το Needle, βρίσκουμε το Fasta Format των ID που θέλουμε να εισάγουμε.

[illegible]

```
# -stdout
# -asequence emboss_needle-I20240501-081009-0580-24884231-p1m.asequence
# -bsequence emboss_needle-I20240501-081009-0580-24884231-p1m.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: JX205496.1
# 2: JX469991.1
# Matrix: EDNAFULL
# Gap_penalty: 10.0
# Extend_penalty: 0.5
#
# Length: 2325
# Identity: 706/2325 (30.4%)
# Similarity: 706/2325 (30.4%)
# Gaps: 1409/2325 (60.6%)
# Score: 935.0
#
#
#=====

JX205496.1      1 ----- 0
JX469991.1      1 ATGGAAGCCTCCGCCGGCTCGTCGCCACCGCACTCCAAGAGAACCGCC 50
JX205496.1      1 ----- 0
```

Έπειτα πάμε στο **Stretch**.

```

# Report_file: stdout
#####
#=====
#
# Aligned_sequences: 2
# 1: JX205496.1
# 2: JX469991.1
# Matrix: EDNAFULL
# Gap_penalty: 10
# Extend_penalty: 1
#
# Length: 2200
# Identity:   782/2200 (35.5%)
# Similarity: 782/2200 (35.5%)
# Gaps:       1159/2200 (52.7%)
# Score: 257
#
#=====
JX205496.1      1 AT-----TC-----
                  ||      ||
JX469991.1      1 ATGGAAGCCTCCGCCGGCTCGTCGCCACCGCACTCCCAAGAGAACCCGCC 50

JX205496.1      5 -----

```

Εδώ βρήκαμε 35.5% ομοιότητα. Άρα βρήκαμε μεγαλύτερη επιτυχία παρόλο που έχοντας λιγότερες παραμέτρους.

Το επόμενο ζητούμενο είναι να μετατρέψουμε ένα fasta αρχείο σε fasta μορφή.

Χρησιμοποιούμε πρώτα το εργαλείο [Sequence conversion website](#) και βλέπουμε ότι μας δίνει το επιθυμητό output.

Fastq to Fasta Sequence Converter

Provided by bugaco.com

Convert file from:

fastq

to

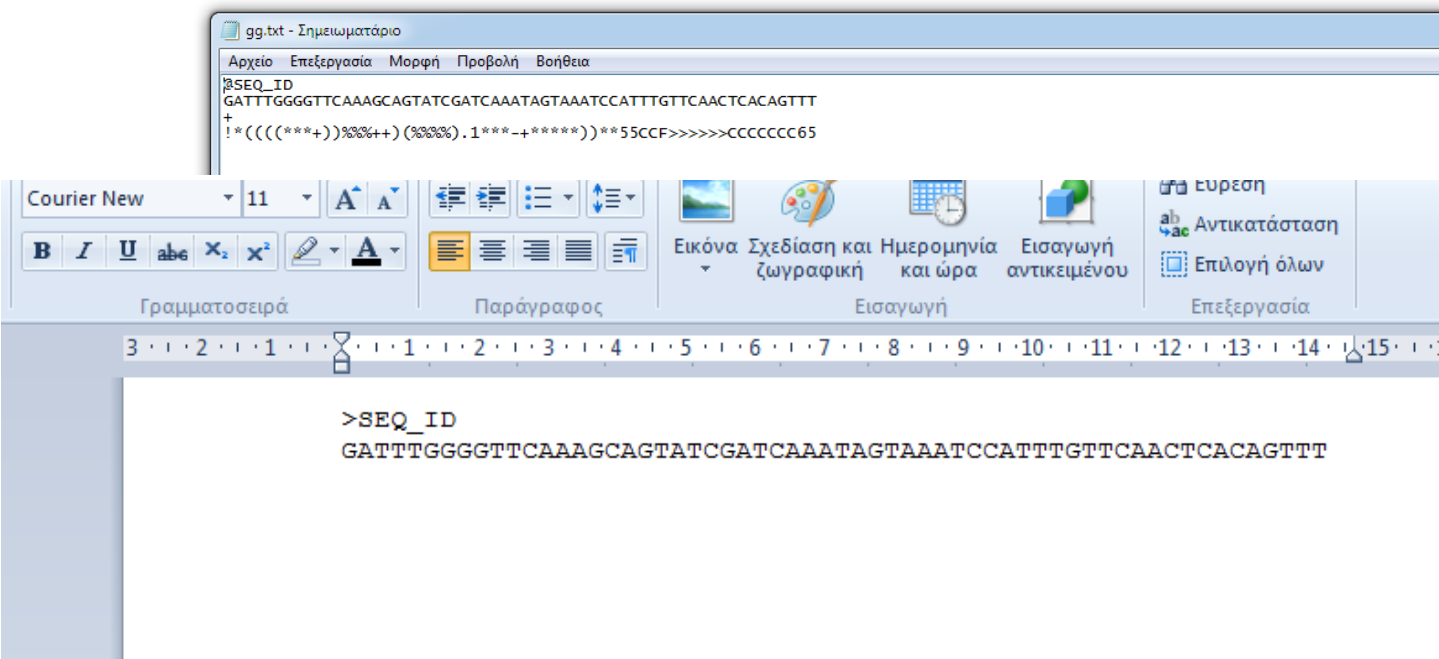
fasta

Alphabet: ☐None ☒DNA ☐RNA ☐Protein ☐Nucleotide

Επιλογή αρχείου gg.txt

Convert

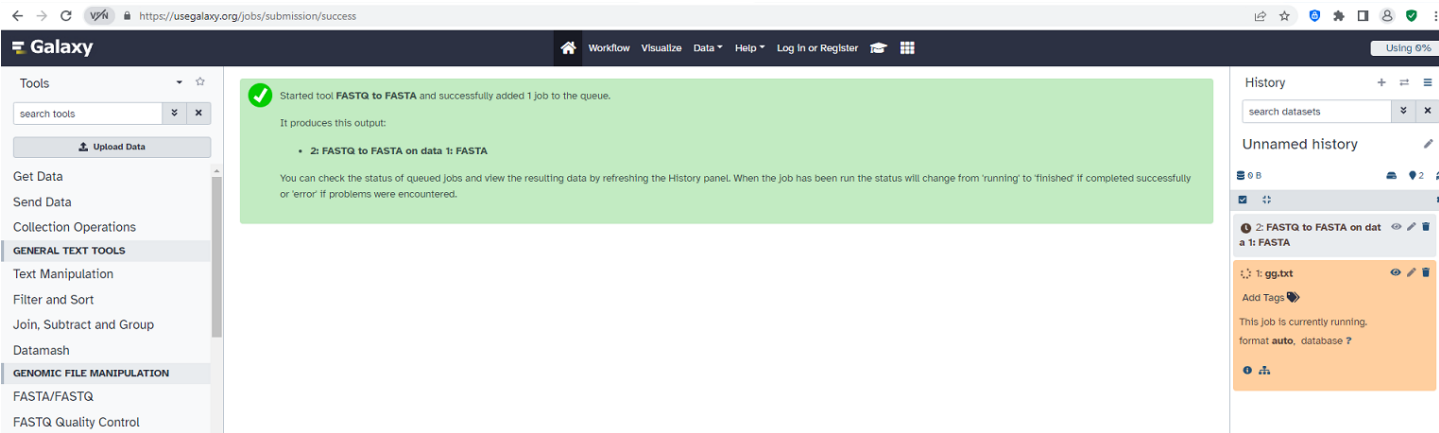
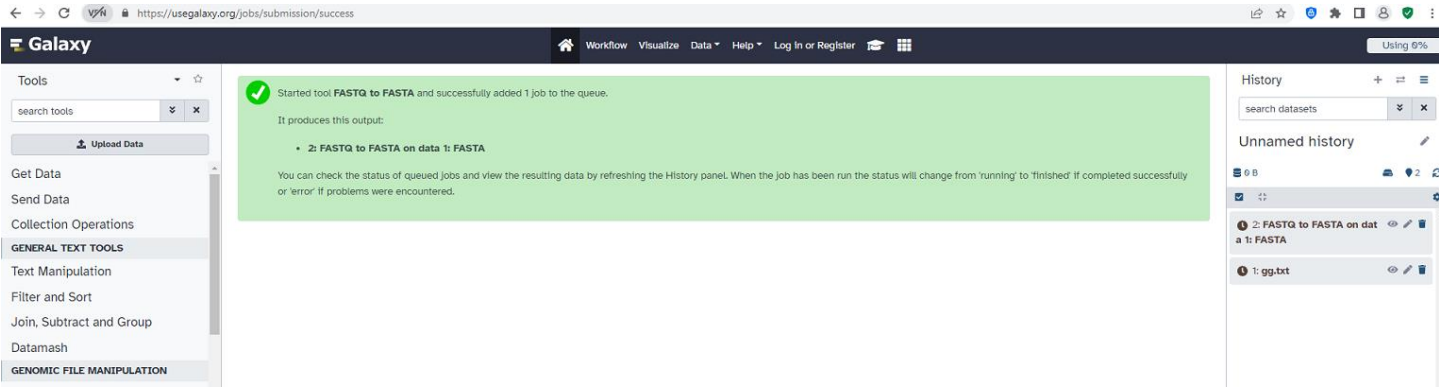
Your file will automatically download when conversion is finished.



Αποφεύγουμε να χρησιμοποιήσουμε το BlastStation διότι απαιτεί λήψη και αγορά.

Θα δοκιμάσουμε όμως με galaxy.

Βλέπουμε ότι δίνει το ίδιο ακριβώς αποτέλεσμα.



History
+
⇌
☰

▼
✕

Unnamed history
✎

196 B
📁
💡 2

☒
⛶

2: FASTQ to FASTA on data 1:
👁
✎
🗑

FASTA

Add Tags
🔖

1 sequences

format **fasta**, database ?

Input: 1 reads.
Output: 1 reads.
discarded 0 (0%) low-quality reads.

📁
🔗
ℹ
🔄
📊
👤
?

>1
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTC

1: gg.txt
👁
✎
🗑

Add Tags
🔖

1 sequences

format **fastqsanger**, database ?

📁
🔗
ℹ
📊
👤

Επίσης θα δοκιμάσουμε και σε biopython.

Input format: **fastq** FASTQ files are a bit like FASTA files but also include sequencing qualities. In Biopython, 'fastq' refers to Sanger style FASTQ files which encode PHRED qualities using an ASCII offset of 33. See also the incompatible 'fastq-solexa' and 'fastq-illumina' variants.

Output format: **fasta** This refers to the input FASTA file format introduced for Bill Pearson's FASTA tool, where each record starts with a '>' line. Resulting sequences have a generic alphabet by default.

How to convert from fastq to fasta ?

You can also convert between these formats by using command line tools.

- On Windows install [WSL](#), on Mac or Linux start terminal
- Install [BioPython](#)
- Run following script:

```
from Bio import SeqIO

records = SeqIO.parse("THIS_IS_YOUR_INPUT_FILE.fastq", "fastq")
count = SeqIO.write(records, "THIS_IS_YOUR_OUTPUT_FILE.fasta", "fasta")
print("Converted %i records" % count)
```

Or you can use this site as online fastq to fasta converter by selecting your formats & file.

[Sequence Converter Home page](#)

Files
main.py
gg copy.fasta
gg.fastq
ggcopy.fasta
main.py

main.py
ggcopy.fasta

1 from Bio import Entrez
2 from Bio import SeqIO
3
4 from Bio import SeqIO
5
6 records = SeqIO.parse("gg.fastq", "fastq")
7 count = SeqIO.write(records, "ggcopy.fasta", "fasta")
8 print("Converted %i records" % count)

ggcopy.fasta
1 >SEQ_ID
2 GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTGTTCACATCAGTTT
3

Format
Run

Converted 1 records

Console
Shell

Run

Converted 1 records

Θα χρησιμοποιήσουμε το εργαλείο fastqc που βρίσκεται online στο site galaxy. Στόχος η ανάλυση ποιότητας ενός fastq αρχείου.

2 · 1 · 1 · 2 · 3 · 4 · 5 · 6 · 7 · 8 · 9 · 10 · 11 ·

@Rosalind_0041
GGCCGGTCTATTTACGTTCTACCCGACGTGACGTACGGTCC
+
6.3536354;.151<211/0?:.6/-2051)-*"40/.,+%)
@Rosalind_0041
TCGTATGCGTAGCACTTGGTACAGGAAGTGAACATCCAGGAT
+
AH@FGGGJ<GB<<9:GD=D@GG9=?A@DC=;;?>839/4856
@Rosalind_0041
ATTCGGTAATTGGCGTGAATCTGTTCTGACTGATAGAGACAA
+
@DJEJEA?JHJ@8?F?IA3=;8@C95=;=?;>D/:;74792.

Δυστυχώς με αυτό το εργαλείο δεν καταφέραμε να δούμε πολλά.

Add Tags

320 lines
format **txt**, database ?

null
Picked up _JAVA_OPTIONS: -

📄 🔗 ℹ️ ↻ 📊 👤 ?

```
##FastQC          0.12.1
>>Basic Statistics      pass
#Measure           Value
Filename           g1g_fastq
File type            Conventional base calls
```

3: g1g.fastq 👁 ✎ 🗑

Add Tags

309 bytes version=1.0
format **fastg**, database ?

uploaded fastg file

📄 🔗 ℹ️ 📊 👤

```
@Rosalind_0041
GGCCGCGTCTATTACGTTCACCCGACGTGACGTACGGTCC
+
6.3536354;.151<211/0?::6/-2051)-*"40/.,+%)
@Rosalind_0041
```

Οπότε θα ψάξουμε και στο Filter FASTQ.

Define Base Offsets as

Absolute Values

Use Absolute for fixed length reads (Illumina, SOLiD)
Use Percentage for variable length reads (Roche/454)

Offset from 5' end *

0

Values start at 0, increasing from the left

Offset from 3' end *

0

Values start at 0, increasing from the right

Aggregate read score for specified range *

mean of scores

Keep read when aggregate score is *

<=

Quality score *

28,0

+ Insert Quality Filter on a Range of Bases

Run Tool

[Help](#)

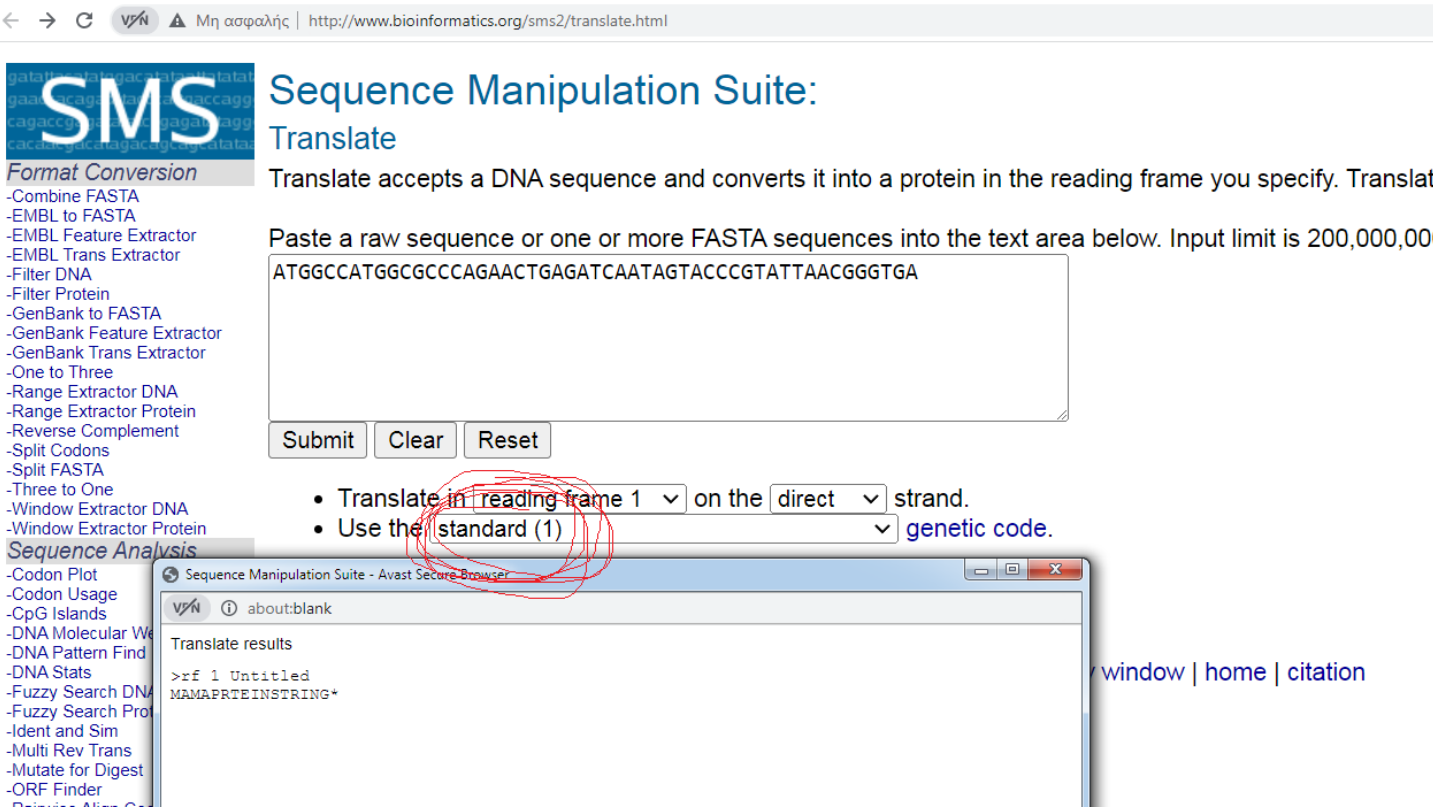


Εδώ βλέπουμε το επιθυμητό αποτέλεσμα.

Τέλος δοκιμάζουμε με biopython.



Τώρα θα μεταφερθούμε στο επόμενο πρόβλημα. Μέσω της σελίδας <http://www.bioinformatics.org/sms2/translate.html> , του εργαλείου μετάφρασης του SMS 2 . Βλέπουμε ότι για την ακολουθία dna που δώσαμε, παράγεται η ζητούμενη πρωτεΐνη με το γεννητικό κώδικα 1. Αυτός ο τρόπος είναι αργός καθώς χρειάζεται να κάνουμε πολλά ερωτήματα μέχρι να πετύχουμε το σωστό κώδικα.

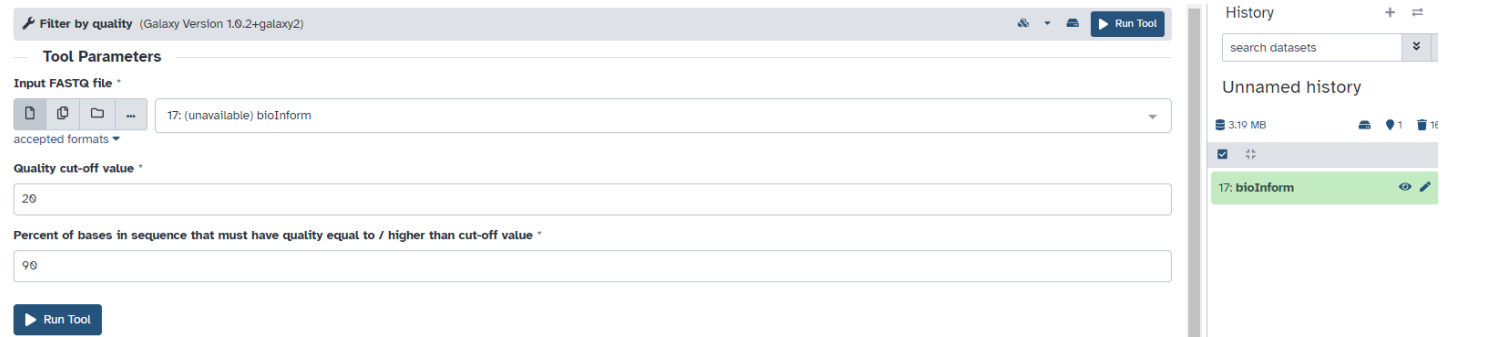


Θα δοκιμάσουμε να λύσουμε το ίδιο πρόβλημα στην biopython.



Τώρα πάμε στο επόμενο πρόβλημα, την εύρεση ποιότητας μέσω του fastg quality filter.

Η απάντηση είναι η αναμενόμενη, 2.



3.19 MB

2

16

18: Filter by quality on data 1

7

Add Tags

242 bytes version=1.0
format **fastg**, database ?

Input: 3 reads.
Output: 2 reads.
discarded 1 (33%) low-quality reads.

@Rosalind_0049_1
GCAGAGACCAAGTAGATGTGTTTGCGGACGGTCGGGCTCCATGTGA
+
FD@@;C<AI?4BA:=>C<G=:AE=><A??>764A8B797@A:58:
@Rosalind_0049_3

17: bioInform

Στο επόμενο πρόβλημα κοιτάμε αν οι 2 ακολουθίες μας είναι ίδιες με τις αντίστροφες συμπληρωματικές τους. Όπως βλέπουμε μόνο η 1 είναι.

Χρησιμοποιήσαμε το [Reverse Complement](#) εργαλείο της sms2. Θα χρησιμοποιήσουμε και biopython.

Sequence Manipulation Suite:

Reverse Complement

Reverse Complement converts a DNA sequence into its reverse, complement, or reverse-complement counterpart. The case of each input sequence character is maintained. You may want to work with the reverse complement of a sequence

Paste the raw sequence or one line at a time

>Rosalind_64
ATAT
>Rosalind_48
GCATA

SubmitClearReset

reverse-complement

*This page requires JavaScript.

*You can [mirror this page](#) or use [our API](#)

Sun 14 Jun 00:37:00 2020

Valid XHTML 1.0; Valid CSS

Sequence Manipulation Suite - Avast Secure Browser

Reverse Complement results
>Rosalind_64 reverse complement
ATAT

>Rosalind_48 reverse complement
TATGC

main.py

main.py > ...

```
1 from Bio.Seq import Seq
2
3 dna_strings = {
4     "Rosalind_64": "ATAT",
5     "Rosalind_48": "GCATA"
6 }
7
8 # Initialize a counter for the matches
9 matches = 0
10
11 # Iterate over the DNA strings
12 for name, dna_string in dna_strings.items():
13     # Create a Seq object for the DNA string
14     my_seq = Seq(dna_string)
15
16     # Get the reverse complement of the DNA string
17     reverse_complement = my_seq.reverse_complement()
18
19     # If the DNA string matches its reverse complement
20     if str(my_seq) == str(reverse_complement):
21         matches += 1
22
23
24 print("Number of DNA strings that match their reverse complements:", matches)
```

Console

Run

Number of DNA strings that match their reverse complements: 1

ΚΩΔΙΚΑΣ

```
from Bio.Seq import Seq

my_seq =
Seq("AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAAGAGTGTCTGATAGCAGC"
)

print("Adenine :" + str(my_seq.count("A")))

print("Cytosine :" + str(my_seq.count("C")))

print("Guanine :" + str(my_seq.count("G")))

print("Thymine :" + str(my_seq.count("T")))

-----

from Bio import Entrez

Entrez.email = "up1067508@upnet.gr"

handle = Entrez.esearch(db="nucleotide", term="Anthoxanthum" + "[Organism] AND (2003/7/25 :
2005/12/27 [Publication Date])")

record = Entrez.read(handle)

print("\n[GenBank gene database]:", record["Count"])

-----

from Bio import SeqIO

from Bio import Entrez

def fetch_and_convert(ids):

    fasta_records = []

    for id in ids:

        handle = Entrez.efetch(db="nucleotide", id=id, rettype="gb", retmode="text")

        record = SeqIO.read(handle, "genbank")

        handle.close()

        # format to FASTA

        fasta_record = record.format("fasta")

        fasta_records.append(fasta_record)

    # return shorter

    return min(fasta_records, key=len)

Entrez.email = "up1067508@upnet.gr"

ids = ["FJ817486", "JX069768", "JX469983"]

print(fetch_and_convert(ids))

-----

from Bio import Entrez

from Bio import SeqIO

Entrez.email = "up1067508@upnet.gr"

handle = Entrez.efetch(db="nucleotide", id=["FJ817486, JX069768, JX469983"], rettype="fasta")

records = list(SeqIO.parse(handle, "fasta")) # Get the list of SeqIO objects in FASTA format

length = [0, 0, 0]

length[0] = len(records[0].seq) # First record ID

length[1] = len(records[1].seq)

length[2] = len(records[2].seq)

last = min(length)

if last == length[0]:

    print(records[0])

elif last == length[1]:

    print(records[1])

elif last == length[2]:
```

```
print(records[2])

-----

from Bio import Entrez

from Bio import SeqIO

from Bio import SeqIO

records = SeqIO.parse("gg.fastq", "fastq")

count = SeqIO.write(records, "ggcopy.fasta", "fasta")

print("Converted %i records" % count)

-----

from Bio import SeqIO

def count_low_quality_reads(fastq_file, quality_threshold):

    count = 0

    for record in SeqIO.parse(fastq_file, "fastq"):

        avg_quality = sum(record.letter_annotations["phred_quality"]) /
len(record.letter_annotations["phred_quality"]) #letter.annotations is quality scores for every base,
stored in the seqrecord

#len -> returns the num items

        if avg_quality < quality_threshold:

            count += 1

    return count

# Test the function with your data

fastq_data = """"@Rosalind_0041

GGCCGGTCTATTACGTTCTCACCCGACGTGACGTACGGTCC

+

6.3536354;.151<211/0?::6/-2051)-*"40/.,+%

)

@Rosalind_0041

TCGTATGCGTAGCACTTGGTACAGGAAGTGAACATCCAGGAT

+

AH@FGGGJ<GB<<9:GD=D@GG9=?A@DC=;::?>839/4856

)

@Rosalind_0041

ATTCGGTAATTGGCGTGAATCTGTTCTGACTGATAGAGACAA

+

@DJEJEA?JHJ@8?F?IA3=;8@C95=;=?;>D/::74792.""

)

with open("fastq_data.fastq", "w") as f:

    f.write(fastq_data)

print(count_low_quality_reads("fastq_data.fastq", 28))

-----

from Bio.Seq import translate

# Given DNA string

dna_string = "ATGGCCATGGCGCCCAGAACTGAGATCAATAGTACCCGTATTAACGGGTGA"

# Loop over the genetic code tables

for gen_table in range(1, 26): # There are 25 known genetic code tables

    protein_string = translate(dna_string, table = gen_table, to_stop=True) # table -> function of translate

    if protein_string == "MAMAPRTEINSTRING":

        print("The index of the genetic code variant used for translation is "+ str(gen_table))

        break

-----

from Bio.Seq import Seq
```

```
dna_strings = {  
    "Rosalind_64": "ATAT",  
    "Rosalind_48": "GCATA"  
}  
  
# Initialize a counter for the matches  
matches = 0  
  
# Iterate over the DNA strings  
for name, dna_string in dna_strings.items():  
    # Create a Seq object for the DNA string  
    my_seq = Seq(dna_string)  
  
    # Get the reverse complement of the DNA string  
    reverse_complement = my_seq.reverse_complement()  
  
    # If the DNA string matches its reverse complement  
    if str(my_seq) == str(reverse_complement):  
        matches += 1  
  
print("Number of DNA strings that match their reverse complements:", matches)
```