Introduction to Bioinformatics

**DNA sequence annotation - Final Project**

**Instructions:**

1. You will work on the project in class and at home.
2. In this project you will work with real data.
3. You will get the DNA sequence by e-mail. The sequences are taken from NCBI website and saved in the text file. You would need to test your program BEFORE you will get your actual sequence for the project.
4. **First, write a function that reads input string and removes all numbers, spaces and new lines from the input string and output one long string of characters without numbers, spaces or new lines.**
5. Please, read the full project description before you start to write your programs. You don’t need to break the project into steps. It was done for your convenience. If you would like to combine steps, you would be able to do it.
6. Read carefully grading rubrics BEFORE you start to work on the project.

**Project Goals:**

* 1. Find all Open Reading Frames (ORF) which is DNA sequence that starts with start codon and ends with one of the STOP codons. These sequencens are potential genes and pseudogenes.
  2. For each potential gene and pseudogene run BLASTX to indentify REAL GENES. ***Real gene is the sequence that encodes the protein.***
  3. For each REAL GENE find the list of organisms this gene is present in. Use Nucleotide BLAST (BLASTn) to perform this task

**Project description**:

**Step 1:** As a first step you will find potential genes (Open Reading Frames) that are present in the input DNA sequence. Your input is one sequence but it includes information from the MAIN strand and from the COMPLEMENT strand. To find genes, follow the rules below.

How to find a gene:

**Rule 1:** Gene starts with codon ATG and ends with one of the following

codons: TAA, TAG, TGA. While calculating the length of the

potential gene in this case, count the start codon, but don’t count an

end codon.

**Rule 2:** If the gene is on the other strand of DNA it will start on the GIVEN

strand with one of the following codons TTA, CTA or TCA as a

“start” codon and a sequence CAT as an “end” codon. While

calculating the length of the potential gene in this case, don’t count

the start codon, but do count an end codon.

**Programming tip:**

Instead of applying this rule directly to the given sequence, you can compliment and reverse the input sequence and to apply rule 1. The rule for finding a DNA compliment is: A → T, T → A, C → G and G → C. Reverse means reading a sequence in the reverse order. For example, if the input DNA sequence is TTACCGTCAT the compliment will be AATGGCAGTA. Reversing the compliment will give a following result: ATGACGGTAA. And now you need to apply rule 1 on the resulting compliment reversed sequence

**Summary and example for treatment of the compliment strand**

1. Your input string includes both strands
2. To find potential genes on the compliment strand follow the following algorithm:
   1. Reverse complement the input string
   2. Apply rule 1 to COMPLIMENT REVERSED STRING to find potential genes and document positions of the start and end codons (ATG and TAA, TAG, TGA)
   3. Go back to the input string and save the sequence between start and end codons you found above. See example

**Input String:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| char | T | T | T | T | A | T | G | T | G | G | C | C | A | A | C | A | T | T | G | C | T | G | C |

**Compliment Reverse String:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| index | 0 | 1 | 2 | 3 | 4 | 5 | **6** | **7** | **8** | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | **18** | **19** | **20** | 21 | 22 |
| char | G | C | A | G | C | A | **A** | **T** | **G** | T | T | G | G | C | C | A | C | A | **T** | **A** | **A** | A | A |

After running Rule 1, your program will find that start codon ATG starts at index 6 and the stop codon, in this case, TAA, starts at index 18 on the COMPLIMENT REVERSE string

These positions are found at the COMPLIMENT REVERSE

START\_CR (ATG) = 6

STOP\_CR (TAA) = 18

**To find corresponded gene on the input string that will include START and STOP codons, apply the following formula:**

START = (Length – 1) – (STOP\_CR + 2) = (23 – 1) – (18 + 2) = 22 – 20 = 2

STOP = (Length – 1) – (START\_CR + 2) = (23 – 1) – (6 + 2) = 22 – 8 = 14

Go back to the input string, and use START and STOP positions you found above to find an actual potential gene sequence. On original input string START position refers to TTA codon and STOP position refers to CAT codon.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| index | 0 | 1 | **2** | **3** | **4** | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | **14** | **15** | **16** | 17 | 18 | 19 | 20 | 21 | 22 |
| char | T | T | **T** | **T** | **A** | T | G | T | G | G | C | C | A | A | **C** | **A** | **T** | T | G | C | T | G | C |

BLAST the potential gene (highlighted in red) to find if the sequence is translated into the protein

**Rule 3:** Additional restriction is that the length of the gene should be

divisible by 3.

**Programming Requirements**:

1. Test your program on the short input sequence, to make sure that the program outputs a correct result and test your program on the real microbes or virus sequence taken from NCBI. Use these links to find testing sequences:

<http://www.ncbi.nlm.nih.gov/genomes/MICROBES/microbial_taxtree.html>

<http://www.ncbi.nlm.nih.gov/genomes/lproks.cgi?view=1>

<http://www.ncbi.nlm.nih.gov/genomes/GenomesGroup.cgi?taxid=10239&opt=Virus&sort=genome>

You can also use this link <http://www.ncbi.nlm.nih.gov/gorf/orfig.cgi>

to find Open Reading Frames (ORF – potential genes) and compare results with your output (not always 100% match, so this testing option is not perfect, but helpful).

1. Input/Output: in this project you will work with the large input data and you would need to read input from the file. You also would need to output the program results into the file. Use UNIX input/output redirection or PYTHON file Input/Output

***The output for this step should include the following:***

1. The list of the potential genes (length larger than 300) in increasing order according to their length in the main string and in the complement string. In step 3 you will finalize if the potential gene is a real gene.
2. The list of the genes that have the length less than 300 (probable pseudogenes) in increasing order according to their length in the main string and in the complement string.
3. Starting position of each potential gene and pseudogene in the input sequence. Assume that the first position of the input sequence is 0 (as usual in Python sequence data type)
4. For your convenience, it would help to summarize the results of this step in four tables: two tables for potential genes (main string and complement string) and two tables for pseudogenes (you can create separate Word file for the summary):

**Tables 1 and 2: Potential genes**

**Table 1 for Main String.**

**Table 2 for Complement String.**

**Length ≥ 300**

**List genes in the increasing order of the length**

|  |  |  |
| --- | --- | --- |
| Gene number | Start position | Length |
| You can assign numbers for the potential genes that you found. You don’t need to copy the actual sub-sequence | Relatively to the beginning of the input sequence |  |

**Table 3 and 4: Pseudogenes**

**Table 3 for Main String.**

**Table 4 for Complement Reverse String.**

**Length ≤ 300**

**List genes in the increasing order of the length**

|  |  |  |
| --- | --- | --- |
| Gene number | Start position | Length |
| You can assign numbers for the potential genes that you found. You don’t need to copy the actual sub-sequence | Relatively to the beginning of the input sequence |  |

**Step 2:** In this step you will BLAST the potentially real genes that you found in step1: (in this step you can use different web resources, for example you can use Biology workbench instead of NCBI).

1. Go to the NCBI home page: <http://www.ncbi.nlm.nih.gov/>
2. Choose BLAST
3. Run BLASTX to identify sequences that are translated to proteins. Use Standard(1) for Genetic Code and UnitProtKB/Swiss-Prot for Database for your initial search. If time permits, try to run BLASTX for additional databases.

***The output for this step should include the following:***

* Save the results of the searches for the final summary.
* It could be that different genes will produce the same results when you are searching protein database.
* For your convenience, it would help to summarize the results of this step in the tables: (you can add an additional column to Tables 1and 2 that you created in step 1):

**Table 1 and 2: Potential genes. Table 1 for Main String. Table 2 for Complement String. Length ≥ 300. List genes in the increasing order of the length**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gene number | Start position | Length | Nucleotide BLAST results | BLASTX results |
| You can assign numbers for the potential genes that you found. You don’t need to copy the actual sub-sequence | Relatively to the beginning of the input sequence |  |  |  |

**Step 3:** In this step you will locate potential promoters in the given DNA sequence for each potential gene that you found in step 1 and find the strength of the promoter. **A promoter is a region of DNA near the beginning of a gene that controls if and when the gene is actually expressed.**

How to find and promoter and its strength:

* 1. For each potential gene that you found in step 1, find a sub-sequence that is located between positions n – 14 and n – 6, including nucleotides at the positing n-14 and n-6, where n is the start position of the potential gene in the input sequence. Pay attention, that n should be larger than 14 for gene to have a promoter. If you a have a potential gene that starts at the position between 0 and 13, this potential gene will not have a promoter. The length of the promoter string is 9
  2. Find an alignment score for the potential promoter you found and the promoter consensus sequence: TG\_TATAAT, where the underscore can be any base. Use the following scoring rule: match = 1 and mismatch = 0. The underscore position always will be considered as a match. The alignment score should be calculated as percent: (score/9)\*100
  3. In general, higher alignment score means better promoter and means that the researched sequence more likely to be a real gene
  4. Repeat the process for the compliment reversed sequence, similar that you did in step 1 of the project

**Final Output for the Project:**

Summarize the results of all steps in the following tables

**Table 1 and 2: Potential genes. Table 1 for Main String. Table 2 for Complement String. Length ≥ 300. LIST GENES IN THE INCRESING ORDER OF THE STRENGTH OF THE PROMOTER**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Gene number | Start position | Promoter Score | Length | Blast Results:  Nucleotide BLAST and BLASTX | Summary and Conclusions |
| You can assign numbers for the potential genes that you found. You don’t need to copy the actual sub-sequence | Relatively to the beginning of the input sequence |  |  |  |  |

**The summary and conclusion part should include your answer if the potential gene could be a real gene based on the strength of the promoter and also BLAST results. You summary should include the finding for main input sequence and for the complement reversed sequence as well.**

**Draw the diagram of the input DNA sequence and positions of potential genes, real genes and pseudogenes.**

**GRADING RUBRICS:**

|  |  |
| --- | --- |
| Grade | Requirements |
| C | Find one potential gene sequence on the main string. Length of the sequence should be larger than 300. BLAST the sequence and document the results of the BLAST and provide a possible summary and conclusion |
| C+ | Find one potential gene sequence on the main string and one potential sequence on the complement string. Length of each sequence should be larger than 300. BLAST each sequence and document the results of the BLAST and provide a possible summary and conclusion |
| B | Find all potential gene sequences on the main string and find all pseudogenes in the main string. BLAST potential genes and document the results of the BLAST and provide a possible summary and conclusion |
| B+ | Find all potential gene sequence on the main string and all potential genes on the complement string. Find all pseudogenes in the main string and on the complement reversed string. BLAST potential genes and document the results of the BLAST and provide a possible summary and conclusion |
| A- | Full completion of the project with minor faults |
| A | Full completion of the project |