Algoritmos Bioinformática / Bioinformática 2021/2022

Finding patterns in sequences

Exercises for class

1. Write a function that finds all the occurrences search_all_occurrences(S, P), of a pattern P in input sequence S. Should return a list of the indices where P occurs in S. Complete in the *sequence_pattern_finding.py* script.

2. Complete the *hamming_distance(s, p)* function.

3. Adapt the previous function to receive as additional input a number of mismatches m (m < len(p)) and returns all the occurrences of p within a distance of m and respective indices.

4. Create a test function that reads from the input a DNA sequence and a pattern to search and finds all its occurrences. Use the function in r*ead_fasta.py* and test with one of the sequences in HBA1.DNA.fasta

5. Write a function that, given a DNA sequence, allows to detect if there are repeated sequences of size k. The result should be a dictionary with sub-sequences as keys, and their frequency as values.

   def repeated_ subsequences_frequency(dna_seq, k = 10):

   …

   Complete in the *sequence_pattern_finding.py* script.

6. In genomic sequences the introns boundaries are defined by sequence signals that can be recognised by their consensus sequences. Consider the splice sites signals as: **GTG** and **CAG***.*
For the given signals find the sequences of all possible introns. The intron sequence will be defined by the contagious stretch between GTG and CAG.

Notes:

- extract the list of indices for the first and sequence signal; operate on the list of indices to find two consecutive indices in the sequence.

- for the analysis ignore the case of letters in the provided file, but use the lower case sequences in file to compare with your results.

Test with HBA1.DNA.fasta and the coronavirus genome sequence.

7. Write a function that given two biological sequences A and B, a pattern size between low and high (integers) where low <= high, finds the frequency of all patterns of size between low and high of A present in B.

def finds_patterns_frequency(seqA, seqB, low, high):

For example consider seqA = "AAATG", seqB ="CAAATC" and low =2 and high = 3

From seqA will tested:

AA (in B)
AA (in B)
AT (in B)
TG (not in B)
AAA (in B)
AAT (in B)
ATG (not in B)

Frequencies will be:

AA - 2
AT - 1
TG - 0
AAA - 1
AAT - 1
ATG - 0

Report only occurring patterns and report them by decreasing order of their frequency.