Bio:522 Algorithms In Computational Biology **Monsoon 2017**

Assignment-3&4: Assembly and Motif
IIIT-DELHI
Due Date: 23rd October 12.00 pm
Total (80 points)
Instructor: Debarka Sengupta

Plagiarism: All submitted codes are expected to be the result of your individual effort. You should never misrepresent someone else's work as your own. In case any plagiarism case is detected you will get one grade reduction in final examination. Cite the resource wherever using other's code.

Instructions:

1) Allowed programing language are python and C . **No external libraries would be allowed**. If you see an ambiguity or inconsistency in a question, please seek a clarification from the teaching staff.

2) You must submit your working solution on backpack on the deadlines page from where you have downloaded this assignment instructions sheet. **No extensions on deadline. If you fail to submit within the time limit then your solution will not be evaluated.**

3) **Mention your enrollment no and name at starting of the each file . Write a clean code with proper comments at appropriate places as it will be checked.Your code should not give error and should not break for nontrivial input and should give warnings wherever needed.**

4) Store the each problem with **rollno_problemX_Y.py** or **rollno_problemX_Y.c** where X is the problem no and Y is task no and upload a zip folder with **rollno_name_assignmentX.zip** containing all the codes and assignment report.**Codes won't be checked if they don't follow the guidelines.**

Hardware and Software Resources:

It is mandatory that you should do version controlling of all your homeworks. We will be using https://bitbucket.org/ for this purpose. Get a login id for free on bitbucket and create the homework repository. All homeworks should be saved inside the repository named hw_1 with your roll number as suffix. If your roll number is 1234 then the repository name will be hw1_1234. Every homework repository should have access level set as "private". You have to use the "share" option for your homework repository to share it with both of your TAs.

Problem 1: Shortest Superstring [10 points]

1] Generate all the synthetic k-mers and then apply shortest superstring algorithm on the obtained k-mers.Do the complexity analysis of the problem and explain in detail.Will you always get the input as the output for any value of k? **[10 points]**

Input:
5
ATACGATATTTAC
Output:
The Shortest Superstring is ATACGATATTTAC

Explanation:
First Line of input k value.
Second Line of Input is input string.
The possible 5-mer for the given input input string are
ATACG,
TACGA,
ACGAT,
CGATA,
GATAT,
ATATT,
TATTT,
ATTTA,
TTTAC
The shortest superstring generated using the 5-mer is **ATACGATATTTAC** which is your final answer.

Problem 2: Graph and Fleury's Algorithm [20 points]

1] Implement Fleury's Algorithm to print eulerian path for a given graph. Explain the time Complexity of the algorithm in brief.Can there be a better algorithmic approach than this ? **[10 points]**

Input:

```
4
4
0 3
0 2
1 2
2 3
```

Output:

```
1-2  2-0  0-3  3-2
```

Explanation:

First line is no of vertex in the graph.Next line is no of edges which is followed by the edges data.

Output is a eulerian path for the given graph.

2] For a connected multi-graph G, G is Eulerian if and only if every vertex has even degree.Prove this statement **[5 points]**

3] A connected graph G is Eulerian if and only if its edge set can be decomposed into cycles.Prove this statement **[5 points]**

Problem 3: De Bruijn Graph [15 points]

1] Consider a set SS of (k+1)-mers of some unknown DNA string. Let Src denote the set containing all reverse complements of the elements of SS.
The de Bruijn graph Bk of order k corresponding to S∪Src is a digraph defined in the following way:

- Nodes of Bk correspond to all k-mers that are present as a substring of a (k+1)-mer from S∪Src.
- Edges of Bk are encoded by the (k+1)-mers of S∪Src in the following way: for each (k+1)-mer r in S∪Src, form a directed edge (r[1:k], r[2:k+1]).

Input: A collection of up to 1000 (possibly repeating) DNA strings of equal length (not exceeding 50 bp) corresponding to a set SS of (k+1)-mers to be read from a file.

Output: The adjacency list corresponding to the de Bruijn graph corresponding to S∪Src

Sample Data:

```
TGAT
CATG
TCAT
ATGC
CATC
CATC
```

Output:

```
(ATC, TCA)
(ATG, TGA)
(ATG, TGC)
(CAT, ATC)
(CAT, ATG)
(GAT, ATG)
(GCA, CAT)
(TCA, CAT)
(TGA, GAT)
```

Problem 4: Planted Motif Search [35 points]

1] Implement any one paper out of the mentioned below:**[25 points]**

- Finding Motifs Using Random Projections by JEREMY BUHLER and MARTIN TOMPA.
- Price, A.; Ramabhadran, S.; Pevzner, P. A. (October 2003). "Finding subtle motifs by branching from sample strings". Bioinformatics
- Pevzner, P. A.; Sze, S. H. (2000). "Combinatorial approaches to finding subtle signals in DNA sequences"
- Pisanti, N.; Carvalho, A.; Marsan, L.; Sagot, M. F. (2006). "Risotto: Fast extraction of motifs with mismatches"
- PMS5: an efficient exact algorithm for the (ℓ, d)- motif finding problem Hieu Dinh, Sanguthevar Rajasekaran* and Vamsi K Kundeti

2] Write a detailed summary report explaining the paper in depth and algorithm you implemented in latex.**[10 points]**