CSE 564

Implementation of DeBruijn Graph

Usage: python Debruijn.py string integer
Example: python Debruijn.py AAABBBA 3
Python Debruijn.py 00011101 3
Note: It works for small string. Due to millions of bases present in DNA sequences, implementation of De Bruijn graph with DNA was not considered. This would have required large memory space.

Input:
String: 00011101
K = 3

Output:

```
 ********************STATISTICS GENERATED************************************
The Total Number of Nodes generated in De Bruijn Graph is 8
All the k-mer generated from user input string is  ['000', '001', '011', '111', '110',
'101']
Balanced nodes in the graph are 11
Balanced nodes in the graph are 10
Node(s) whose indegree#outdegree in the graph is/are 00
Node(s) whose indegree#outdegree in the graph is/are 01
The path covers these nodes ['01', '11', '11', '10', '00', '01']
The path covers these edges only once, following Eulerian path ['101', '011', '111',
'110', '000', '001']
```

Input :
String = TAATGCCATGGGATGTT
K=3

Output:

```
********************STATISTICS GENERATED************************************
The Total Number of Nodes generated in De Bruijn Graph is 20
All the k-mer generated from user input string is  ['TAA', 'AAT', 'ATG', 'TGC', 'GCC',
'CCA', 'CAT', 'ATG', 'TGG', 'GGG', 'GGA', 'GAT', 'ATG', 'TGT', 'GTT']
Balanced nodes in the graph are AA
Balanced nodes in the graph are GT
Balanced nodes in the graph are CC
Balanced nodes in the graph are CA
Balanced nodes in the graph are GG
Balanced nodes in the graph are GC
Balanced nodes in the graph are GA
Node(s) whose indegree#outdegree in the graph is/are AT
Node(s) whose indegree#outdegree in the graph is/are TG
The path covers these nodes ['AT', 'TT', 'CA', 'AT', 'CC', 'TG', 'AT', 'AA', 'GG', 'GA',
'GG', 'GT', 'GC']
The path covers these edges only once, following Eulerian path ['AAT', 'GTT', 'CCA',
'CAT', 'GCC', 'ATG', 'GAT', 'TAA', 'GGG', 'GGA', 'TGG', 'TGT', 'TGC']
```