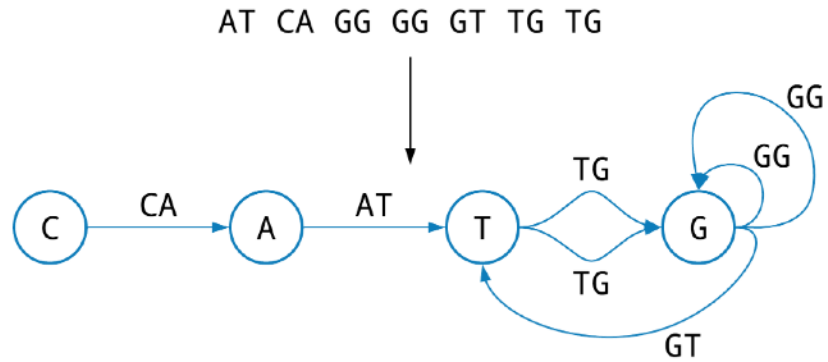## 3E   Construct the De Bruijn Graph of a Collection of k-mers

De Bruijn Graph from k-mers Problem

Construct the de Bruijn graph from a collection of k-mers.

Input: A collection of k-mers *Patterns*.
Output: The graph DEBRUIJN(*Patterns*).

Formatting

Input: A space-separated list of k-mer strings *Patterns*.
Output: An adjacency list representing DEBRUIJN(*Patterns*).

Constraints

- The number of patterns in the string-set *Patterns* will be between 1 and 10^4.

- The length of any one pattern in *Patterns* will be between 1 and 10^2.

- All strings in *Patterns* will be DNA strings.

Test Cases

Case 1

Description: The sample dataset is not actually run on your code.

Input:
```
GAGG CAGG GGGG GGGA CAGG AGGG GGAG
```

Output:
```
GAG : AGG
CAG : AGG AGG
GGG : GGG GGA
AGG : GGG
GGA : GAG
```

Case 2

Description: The sample dataset is not actually run on your code.

Input:
```
GCAAG CAGCT TGACG
```

Output:
```
GCAA : CAAG
CAGC : AGCT
TGAC : GACG
```

Case 3

Description: The sample dataset is not actually run on your code.

Input:
```
AGGT GGCT AGGC
```

Output:
```
AGG : GGT GGC
GGC : GCT
```

Case 4

Description: The sample dataset is not actually run on your code.

Input:
```
TTCT GGCT AAGT GGCT TTCT
```

Output:
```
TTC : TCT TCT
GGC : GCT GCT
AAG : AGT
```

Case 5

Description: The sample dataset is not actually run on your code.

Input:
```
CA CA CA CA CC CA
```

Output:
```
C : A A A A C A
```

Case 6

Description: A larger dataset of the same size as that provided by the randomized autograder. Check input/output folders for this dataset.