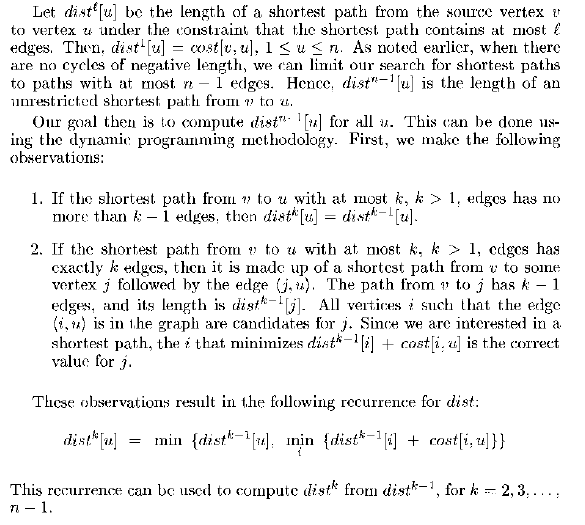
Experiment 6 2nd May 2022

**Single-source shortest path: General weights**

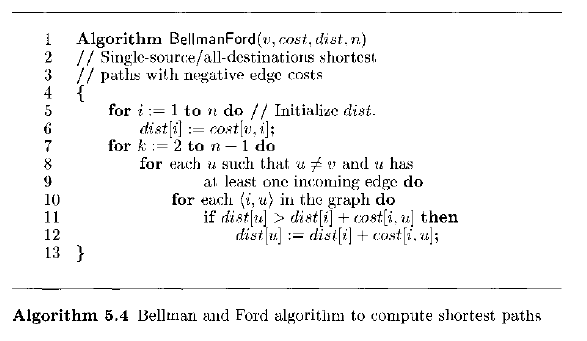
Aim: To write a program to implement the single-source shortest path algorithm using Dynamic Programming.

Theory:

Consider the single-source shortest path problem when some or all the edges of the directed graph G may have negative length. When negative edge lengths are permitted, we require that the graph have no cycles of negative length. This is necessary to ensure that shortest paths consist of a finite number of edges. When there are no cycles of negative length, there is a shortest path between any two vertices of an n-vertex graph that has at most n-1 edges on it. 

Task:

Write a program to implement the following algorithm.



Program and Output:

Input the Cost Adjacency Matrix

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Vertex | 1 | 2 | 3 | 4 | 5 |
| 1 | 0 | 6 | 7 | ∞ | ∞ |
| 2 | ∞ | 0 | 8 | 5 | -4 |
| 3 | ∞ | ∞ | 0 | -3 | 9 |
| 4 | ∞ | -2 | ∞ | 0 | ∞ |
| 5 | 2 | ∞ | ∞ | 7 | 0 |

Output the distance matrix distk[1…5]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| k | 1 | 2 | 3 | 4 | 5 |
| 1 | 0 | 6 | 7 | ∞ | ∞ |
| 2 | 0 | 4 | 7 | 4 | 2 |
| 3 | 0 | 2 | 7 | 4 | 2 |
| 4 | 0 | 2 | 7 | 4 | -2 |
| 5 | 0 | 2 | 7 | 4 | -2 |

Conclusion: