Name: K. Avashya
USN: 1BM17CS034
Lab 7.

Q WAP for distance vector algorithm to find path for transmission

```
Class distvector:
def __init__(self, n):
self.matrix = []
self.n = n
def newEdge (self, u, v, w):
self.matrix.append((u, v, w))
def display (self, dist, src):
print("vector table of {}", format
                (chr(ord('A') + src)))
for i in range (self.n):
    print("\t {0} \t {1}".format(chr(ord
                ('A') + i), dist[i]))
def start(self, src):
    dist = [90] self.n
    dist [src] = 0
    for - in range(self.n-1):
for u, v, w, in Self.matrix:
if dist[v] != 99 and dist[u] + w <
                                dist[v]:
    dist[v] = dist[u] + w
    Self.display(dist, src)
matrix = []
print("Enter the routers")
n = int(input)
print("Enter the adjacency matrix
        (enter the 999 for infinity):")
    for i in range(n):
```

```
m = list(map(int, input().split(" ")))
matrix.append(m)
g = distVector(n)
for i in range(n):
    for j in range(n):
        if matrix[i][j] == 1:
            g.newEdge(i, j, 1)
for _ in range(n):
    g.start(_)
```