MP-1 PRACTICAL-6

1. Develop a python program to demonstrate the Initial Basic Solution in Transportation problem using Vogel method in Linear Programming (U-V method).

Code:

```
from collections import defaultdict
costs  = {'W': {'A': 16, 'B': 16, 'C': 13, 'D': 22, 'E': 17},
        'X': {'A': 14, 'B': 14, 'C': 13, 'D': 19, 'E': 15},
        'Y': {'A': 19, 'B': 19, 'C': 20, 'D': 23, 'E': 50},
        'Z': {'A': 50, 'B': 12, 'C': 50, 'D': 15, 'E': 11}}
demand = {'A': 30, 'B': 20, 'C': 70, 'D': 30, 'E': 60}
cols = sorted(demand.iterkeys())
supply = {'W': 50, 'X': 60, 'Y': 50, 'Z': 50}
res = dict((k, defaultdict(int)) for k in costs)
g = {}
for x in supply:
   g[x] = sorted(costs[x].iterkeys(), key=lambda g: costs[x][g])
for x in demand:
   g[x] = sorted(costs.iterkeys(), key=lambda g: costs[g][x])

while g:
   d = {}
   for x in demand:
      d[x] = (costs[g[x][1]][x] - costs[g[x][0]][x]) if len(g[x]) > 1 else costs[g[x][0]][x]
   s = {}
   for x in supply:
      s[x] = (costs[x][g[x][1]] - costs[x][g[x][0]]) if len(g[x]) > 1 else costs[x][g[x][0]]
   f = max(d, key=lambda n: d[n])
   t = max(s, key=lambda n: s[n])
   t, f = (f, g[f][0]) if d[f] > s[t] else (g[t][0], t)
   v = min(supply[f], demand[t])
   res[f][t] += v
   demand[t] -= v
   if demand[t] == 0:
      for k, n in supply.iteritems():
         if n != 0:
            g[k].remove(t)
      del g[t]
      del demand[t]
   supply[f] -= v
   if supply[f] == 0:
      for k, n in demand.iteritems():
         if n != 0:
            g[k].remove(f)
```

```
      del g[f]
      del supply[f]

for n in cols:
   print "\t", n,
print
cost = 0
for g in sorted(costs):
   print g, "\t",
   for n in cols:
      y = res[g][n]
      if y != 0:
         print y,
      cost += y * costs[g][n]
      print "\t",
   print
print "\n\nTotal Cost = ", cost
```

Output:-

```
        A       B       C       D       E
W                       50
X       30              20              10
Y               20              30
Z                                       50


Total Cost =  3100
```