LAB. TEST -2	IBM 18CSOLL
Si isana sana sana sana sana sana sana sa	
() and if the total	
Program 3	54
The state of the state of the	47
class Topology:	
del int - (sell array of spints):	86 64 T = 1560N
def init - (self, array of points): self nodes = array - of - point self. edges = [7]	is a second of the second of t
911. edoes = [7	house analogon a in
A W	
de Malacran tout the le	p2 cost): it is.
def add-direct connection (self, pt, self edges append ((pt, p2, cost)) self edges append ((p2, p1, cost))	Harmon Feel Man
coll of a second ((s) of cost)	
and code about a (Cbr. b., cost)	
del licharia ser tra ma li (coll).	
def distance-vector-vouting (self): import collections	
	* SALAN SIMPLE
for note in self notes:	- L 1- L)
dist = collections defaultdi	7
next-hop = { node: node	
for other node in sell	-nodes:
if other-node != nod	NO.0000
dist [other node] =	499999
for i in vange (sen (self no	odes)-1):
for edge in self.eda	ys:
SVC, deet coet = e	dge
if dist[src] + cost <	dist[dest]:
dist[dest] = dist[s	irc] + cost
if src == node:	
next_hop[dest	7 = dest
elif suc in next-hop(dest]	= next has for?
, rest-poplars	- 1001 > 1001 5101

f print_routing_table(self, node, dist_next_hop):

print (f Routing table for {node}:) }

print (Doet it Cost of Next hop)

for dest, (ost in dist. items():

print(f Edust) it {costy it {next-hop/dest]}') nodes = ['A' B' find' D' D' E' - for find of t. distance-vector-vouting () indial