b-)

func BellmanFord(firstvertex, aimedvertex)

for each vertex V in Graph

dist [V] <- ∞

dist [firstvertex] <- 0

for each vertex V in G

for each u,v,w in G

If dist[u] + w < dist[V] and dist[u] != infinity

dist [v] <-dist [u] + w

return minimum(dist [firstvertex], dist [aimedvertex +totalnumofvertex])

func frombustotrain()

for each V/2 in range of totalnumofvertex

addedge to bus graph firstvertex to firstvertex + V/2 weight with 0

fucntion fromtraintobus()

for each V/2 in range of totalnumofvertex

addedge to train graph firstvertex to firstvertex + V/2 weight with 0

shorter\_length <- minimum(traingraph.BellmanFord, busgraph.BellmanFord)

d-)

