1) Initialization{G,S}: { for each vertex v
v.d = ∞, v.π = nil
s.d = 0

Q = Heap { Build Heap(List nodes){
for nodes/2 to 0{
minHeapify(nodes, i){ 3 { check against Child
nodes. If > Child, mov
down

While Q is not Empty{
V = ExtractMin(Q)
for Each edge (U,u):
Relax (U,u) { if u.d > v.d + w(U,u)
u.d = v.d + w(U,u)
u.π = v

4) $O(u) + \sum_{\text{node } v} O(\log(u)) + mv\, O(\log(n))$

2 Cases: $O(n \log n)$

2) Adjacency Matrix: $O(V^2 + V \log V) = O(V^2)$

If Matrix Used.