Algorithm 1 Tabu Search

1: bestStructure, s ← random_conformation()
2: bestCost ← cost(bestStructure)
3: while not stop() do
4:     N ← compute_neighbours(s)
5:     sort N with respect to cost
6:     for all i ∈ N do
7:         if cost($N_i$) < bestCost then
8:             bestCost ← cost($N_i$)
9:             s, bestStructure ← $N_i$
10:            break loop
11:        end if
12:        if not Tabu($N_i$, Q) then
13:            s ← $N_i$
14:            break loop
15:        end if
16:    end for
17:    pushback( Q, s )
18: end while
19: return bestStructure