Algorithmique de base

Master 1, Université de Rennes

26/09/2024 – TD 4

Exercise 1 (Proof of correctness of BFS)**.** Let $G = (V, E)$ be a directed or undirected graph. Recall that $\delta(x, y)$ denotes the minimum number of edges in any path from vertex x to vertex y.

1. Let $s \in V$ be an arbitrary vertex. Show that for any edge $(u, v) \in E$, $\delta(s, v) \leq \delta(s, u) + 1$.
2. Suppose that BFS is run on G from a given source vertex $s \in V$. Show that upon termination, for each vertex $v \in V$, the value $v.d$ computed by BFS satisfies $v.d \geq \delta(s, v)$.

   Hint: Use induction on the number of ENQUEUE operations. The base case of the induction is the situation immediately after enqueuing s in line 10 of BFS. We have $s.d = 0 = \delta(s, s)$ and $v.d = \infty \geq \delta(s, v)$ for all $v \in V \setminus 0\{s\}$.
3. Suppose that during the execution of BFS,the queue Q contains the vertices $v_1, \ldots, v_r$, where v_1 is the head of Q and v_r is the tail. Show that $v_r.d \leq v_1.d$ and $v_i.d \leq v_{i+1}.d$ for all $i = 1, 2, \ldots, r - 1$.

   Hint: Use induction on the number of queue operations. When the queue contains only s, the statement is clearly true. For the inductive step, we prove that it holds after both dequeuing and enqueuing a vertex.
4. Suppose that vertices v_i and v_j are enqueued during the execution of BFS, and that v_i is enqueued before v_j. Show that $v_i.d \leq v_j.d$ at the time that v_j is enqueued.
5. Assume that some vertex v receives a d value not equal to its shortest-path distance.
   (a) Show that $v.d > \delta(s, v)$.
   (b) Let u be the vertex immediately preceding v on a shortest path from s to v. Show that $v.d > u.d + 1$.
   (c) Consider the moment when BFS chooses to dequeue vertex u from Q in line 12. In this moment, vertex v is either white, gray, or black. Show that in all this cases $v.d \leq \delta(s, v)$
6. Conclude that BFS discovers every vertex $v \in V$ that is reachable from the source s, and upon termination, $v.d = \delta(s, v)$ for all $v \in V$.

Algorithm 1 BFS

```
function BFS(G, s):
    for each vertex u ∈ G.v \ {s} do:
        u.color = WHITE
        u.d = ∞
        u.π =NIL
    s.color = GRAY
    s.d = 0
    s.π =NIL
    Q = ∅
    ENQUEUE(Q, s)
    while Q ≠ ∅ do:
        u =DEQUEUE(Q)
        for each vertex v ∈ G.Adj[u] do:
            if v.color ==WHITE then:
                v.color =GRAY
                v.d = u.d + 1
                v.π = u
                ENQUEUE(Q, v)
        u.color =BLACK
```