Breadth-First-Search (BFS)

Generic Search Alg:
1. Exp = \{s\} Exp = set of explored nodes
2. While (\exists u, v \in E: (u \in Exp \land v \notin Exp))
3. Add v to Exp

Big Question:
If multiple edges cross boundary between explored and unexplored, which do explore first?

One strategy: explore all edges crossing current boundary, then look at new boundary & explore
Shortest Path BFS Page 2

S. Kimmel

Idea, explore layers.

Exp[v] = 0 \quad \forall v \in V \quad \text{// mark True when } \text{"explored"}

A = \{s\};
A \cdot \text{add}(s);
Exp[s] = \text{True}

A is a queue = "First in, first out"

while (A is not empty) {
    v = A.pop
    \text{For each edge } (v, w) \quad \text{and}
    \quad \text{if } (\text{Exp}[w] = \text{false}) \quad \text{then}
    \quad \quad \text{Exp}[w] = \text{True}; \quad A \cdot \text{add}(w);
    \text{end if}
    \text{end for}
}

This is Breadth First Search - slowly move away in layers from initial node.
Q: What is runtime of BFS using adjacency list if n is total # of vertices, m is total # of edges, n_s is # of vertices reachable from s, m_s is # of edges reachable from s.

A. \(O(m_s)\)  
B. \(O(n+m_s)\)  
C. \(O(n_s \cdot m_s)\)  
D. \(O(n+n_s \cdot m_s)\)

Answer: B.

1: Initializing \(\text{Exp}[v] \forall v \in V\) takes time \(O(n)\)