PSET 3, Problem 3

Will Childs-Klein {wdc22}

3

```
Given: L,n,m

def isValid(D):
  if D(m, n) > L
    return False
  return search(D, 1, 1)
end def

def search(D, i, j):
  if D(i, j) < L
    return False
  else if i == m and j == n
    return True
  else
    down = i+1
    right = j+1
    return search(D, down, j) OR search(D, i, right)  // note: incluse OR
end def
```

Runtime

This algorithm runs in $O(2mn)$. The algorithm executes a breadth-first search on the m x n matrix presented by D. As BFS is bounded by the number of edges in the graph (in this case, $|E| = m$ x n), the runtime of the algorithm is polynomial in m, n, as each node can have max 2 edges (right or down, i.e. a moves or b moves from one platform to the next).

Correctness

The algorithm is correct because it only considers moves of $<$ L as valid in the search. Because the algorithm will only "move" to a platform (i,j) with D(i,j) of

<= L, it will either find the end plafor (m,n) or return false. The algorithm also checks if platform (1,1) is valid before recursing.