PSET 3, Problem 4

Will Childs-Klein {wdc22}

4

```
def isValid(arr M):
  init: arms a,b
        a.pos      = null   a.t      = null
        b.pos      = null   b.t      = null
  if n <= 2
    return True

  for i = [1,n]:
    a_dist = dist(i, M[i], a.t, a.pos)
    b_dist = dist(i, M[i], b.t, b.pos)

    if  a_dist < b_dist
      a.pos = M[i]
      a.t = i
    else if b_dist < a_dist
      b.pos = M[i]
      b.t = i
    else if a_dist == b_dist && a_dist < infiniti
      // either a or b would work
      a.pos = M[i]
      a.t = i
    else
      return False
  end for

  return True
end def

def dist(i, note, last_i, pos):
  if pos == null && t == null
    return 0
  else if |pos - note| > |t - i|
    return infiniti
```

```
  else
    return |pos - note|
end def
```

Runtime

This algorithm runs in $O(n)$ with n as the lenth of the musical piece

Correctness

this algorithm isn't correct. upon encountering a note out of range of both a and b, it doesn't back-track through its move history to see whether or not a note to which the robot was indifferent in arm choice (i.e. either arm could have reached the note), let's say a, could have been played by the other arm, let's say b, and then roll-back its execution to that point, proceeding down the new branch.