Algorithm 1 rand_center(qsource, qgoal, map_size, obstacles)

function RAND_CENTER($qsource, qgoal, map\_size, obstacles$)
  $straightLineMidpoint \leftarrow [\frac{(qsource(1)+qgoal(1))}{2} \frac{(qsource(2)+qgoal(2))}{2}]$
  $straightLineStartX \leftarrow linspace(qsource(1), straightLineMidpoint(1), \max(100, map\_size))$
  $straightLineStartY \leftarrow linspace(qsource(2), straightLineMidpoint(2), \max(100, map\_size))$
  $straightLineStartX \leftarrow linspace(qgoal(1), straightLineMidpoint(1), \max(100, map\_size))$
  $straightLineStartY \leftarrow linspace(qgoal(2), straightLineMidpoint(2), \max(100, map\_size))$
  $validStraightLineFromStart \leftarrow CheckCollision(straightLineFromStart, obstacles, map\_size)$
  $validStraightLineFromEnd \leftarrow CheckCollision(straightLineFromEnd, obstacles, map\_size)$
  $centerPointsFromStart \leftarrow$floor(0.25*height(validStraightLineFromStart))
  $centerPointsFromEnd \leftarrow$floor(0.25 *height(validStraightLineFromEnd))
  $centerStartPoints \leftarrow zeros(numberCenterPointsFromStart, 2)$
  for $i = 1 \rightarrow centerPointsFromStart$ **do**
    $centerStartPoints(i, :) \leftarrow validStraightLineFromStart(i)$
  end for
  $centerEndPoints \leftarrow zeros(numberCenterPointsFromEnd, 2)$
  for $i = 1 \rightarrow numberCenterPointsFromEnd$ **do**
    $centerEndPoints(i, :) \leftarrow validStraightLineFromEnd(i)$
  end for
  $sampleSet \leftarrow centerPoints$
  $point \leftarrow sampleSet(randi(height(sampleSet), 1), :)$
  return *point*
end function

Algorithm 2 rand_center(qsource, qgoal, map_size, obstacles)

function RAND_CENTER($qsource, qgoal, map\_size, obstacles$)
  $straightLineMidpoint \leftarrow [\frac{(qsource(1)+qgoal(1))}{2} \frac{(qsource(2)+qgoal(2))}{2}]$
  $straightLineStartX \leftarrow linspace(qsource(1), straightLineMidpoint(1), \max(100, map\_size))$
  $straightLineStartY \leftarrow linspace(qsource(2), straightLineMidpoint(2), \max(100, map\_size))$
  $straightLineStartX \leftarrow linspace(qgoal(1), straightLineMidpoint(1), \max(100, map\_size))$
  $straightLineStartY \leftarrow linspace(qgoal(2), straightLineMidpoint(2), \max(100, map\_size))$
  $validStraightLineFromStart \leftarrow CheckCollision(straightLineFromStart, obstacles, map\_size)$
  $validStraightLineFromEnd \leftarrow CheckCollision(straightLineFromEnd, obstacles, map\_size)$
  $centerPointsFromStart \leftarrow$floor(0.25*height(validStraightLineFromStart))
  $centerPointsFromEnd \leftarrow$floor(0.25 *height(validStraightLineFromEnd))
  $centerStartPoints \leftarrow zeros(numberCenterPointsFromStart, 2)$
  for $i = 1 \rightarrow centerPointsFromStart$ **do**
    $centerStartPoints(i, :) \leftarrow validStraightLineFromStart(i)$
  end for
  $centerEndPoints \leftarrow zeros(numberCenterPointsFromEnd, 2)$
  for $i = 1 \rightarrow numberCenterPointsFromEnd$ **do**
    $centerEndPoints(i, :) \leftarrow validStraightLineFromEnd(i)$
  end for
  $sampleSet \leftarrow centerPoints$
  $point \leftarrow sampleSet(randi(height(sampleSet), 1), :)$
  return *point*
end function