

```

// @param {number} radius - radius of the circle
// @param {number} x - x coordinate of the center
// @param {number} y - y coordinate of the center
// @return {boolean} true if the point is inside the circle, false otherwise
function isPointInCircle(radius, x, y) {
  // Calculate the distance from the center to the point
  const distance = Math.sqrt((x - x0) ** 2 + (y - y0) ** 2);
  // Check if the distance is less than or equal to the radius
  return distance <= radius;
}

// Example usage:
// isPointInCircle(5, 10, 10) // true
// isPointInCircle(5, 15, 15) // false

```

1. **Function Definition:** The function `isPointInCircle` takes three arguments: `radius` (a number), `x` (the x-coordinate of the center), and `y` (the y-coordinate of the center).

2. **Distance Calculation:** Inside the function, the distance from the center point  $(x_0, y_0)$  to the point  $(x, y)$  is calculated using the Euclidean distance formula:  $\text{distance} = \sqrt{(x - x_0)^2 + (y - y_0)^2}$ .

3. **Comparison:** The calculated distance is compared to the radius. If the distance is less than or equal to the radius, the point is inside the circle.

4. **Return Statement:** The function returns a boolean value: `true` if the point is inside the circle, and `false` otherwise.

5. **Example Usage:**

- `isPointInCircle(5, 10, 10)` returns `true` because the point  $(10, 10)$  is at the center of a circle with radius 5.
- `isPointInCircle(5, 15, 15)` returns `false` because the point  $(15, 15)$  is outside a circle with radius 5 centered at  $(10, 10)$ .