Problem 478: Generate Random Point in a Circle

Problem Information

Difficulty: Medium
Acceptance Rate: 42.06%
Paid Only: No
Tags: Math, Geometry, Rejection Sampling, Randomized

Problem Description

Given the radius and the position of the center of a circle, implement the function `randPoint` which generates a uniform random point inside the circle.

Implement the `Solution` class:

* `Solution(double radius, double x_center, double y_center)` initializes the object with the radius of the circle `radius` and the position of the center `(x_center, y_center)`. * `randPoint()` returns a random point inside the circle. A point on the circumference of the circle is considered to be in the circle. The answer is returned as an array `[x, y]`.

Example 1:

Input ["Solution", "randPoint", "randPoint", "randPoint"] [[1.0, 0.0, 0.0], [], [], []] **Output** [null, [-0.02493, -0.38077], [0.82314, 0.38945], [0.36572, 0.17248]] **Explanation** Solution solution = new Solution(1.0, 0.0, 0.0); solution.randPoint(); // return [-0.02493, -0.38077] solution.randPoint(); // return [0.82314, 0.38945] solution.randPoint(); // return [0.36572, 0.17248]

Constraints:

* `0 < radius <= 108` * `-107 <= x_center, y_center <= 107` * At most `3 * 104` calls will be made to `randPoint`.

Code Snippets

C++:

```
class Solution {
public:
Solution(double radius, double x_center, double y_center) {

}

vector<double> randPoint() {

}
};

/**
* Your Solution object will be instantiated and called as such:
* Solution* obj = new Solution(radius, x_center, y_center);
* vector<double> param_1 = obj->randPoint();
*/
```

Java:

```
class Solution {

public Solution(double radius, double x_center, double y_center) {

}

public double[] randPoint() {

}
}

/**
* Your Solution object will be instantiated and called as such:
* Solution obj = new Solution(radius, x_center, y_center);
* double[] param_1 = obj.randPoint();
*/
```

Python3:

```
class Solution:

def __init__(self, radius: float, x_center: float, y_center: float):


def randPoint(self) -> List[float]:



# Your Solution object will be instantiated and called as such:
# obj = Solution(radius, x_center, y_center)
# param_1 = obj.randPoint()
```