

Parcial II Término 2004-2005. Diciembre 2004

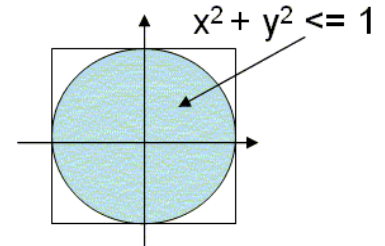
Tema 3. (25 puntos) Encuentre un valor aproximado de la constante π con el siguiente procedimiento. Considere un círculo de radio unitario, centrado en el origen e inscrito en un cuadrado:

Dado un valor n , genere las coordenadas x , y para n puntos.

Asigne valores aleatorios reales entre 0 y 1 y cuente cuantos puntos caen dentro del cuadrante de círculo.

Si llamamos a este contador k , se puede establecer la siguiente relación aproximada suponiendo n grande:

$$\frac{k}{n} = \frac{(1/4 \text{ del área del círculo})}{(1/4 \text{ del área del cuadrado})} = \frac{(1/4) \pi (1)^2}{(1/4) (2)^2}$$



Donde se puede obtener el valor aproximado de π .

Rúbrica: Manejo de aleatorios (5 puntos), puntos en el círculo (5 puntos), repetir n veces (10 puntos). Solución integral (5 puntos)


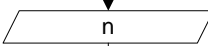
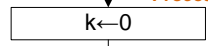
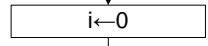

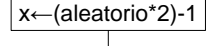
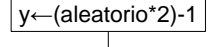
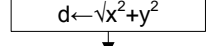
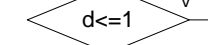
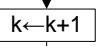
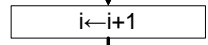
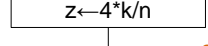
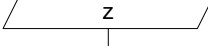

Propuesta de Solución:

Ingresar n como la cantidad de puntos en el plano a ubicar de forma aleatoria dentro el rango del cuadrado que inscribe al círculo.

Usar una variable " k " como el contador para los puntos que caen dentro del círculo. Al generar cada punto se puede calcular la distancia al centro mediante Pitágoras.

Se repite el proceso para n puntos y al final se calcula el valor estimado de π acorde con la relación presentada.

Nota: no se usa como variable " π " debido a que es variable reservada.

Descripción	DIAGRAMA DE FLUJO	Python
Inicio		<code># ICM00794-Fundamentos de Computación - FCNM-ESPOL</code>
Ingresar los n puntos para estimar el valor de " π "		<code># Parcial II Término 2004</code>
Contador puntos en el círculo		<code># Tema3. Calcula Pi Montecarlo</code>
Contador aleatorios generados		<code># Propuesta de solución. edelros@espol.edu.ec</code>
Mientras existan puntos por generar		<code># Considera todo el círculo</code>
Coordenada aleatoria x		<code>import random</code>
Coordenada aleatoria y		<code>import math</code>
Distancia del punto al origen		<code>n=int(input('¿Cuántos puntos?: '))</code>
Si la distancia es menor que el radio		<code>#Procedimiento</code>
Se cuenta un punto dentro del círculo		<code>k=0</code>
Cuenta punto generado		<code>i=0</code>
Repita		<code>while (i<n):</code>
Estimación de valor de π		<code>x=(random.random()*2)-1</code>
Mostrar resultado de π		<code>y=(random.random()*2)-1</code>
		<code>d=math.sqrt(x**2+y**2)</code>
		<code>if (d<=1):</code>
		<code>k=k+1</code>
		<code>i=i+1</code>
		<code>z=4*k/n</code>
		<code>#Salida</code>
		<code>print('Pi estimado es:')</code>
		<code>print(z)</code>

Ejecución del algoritmo: montecarlopi.py

<pre>>>> ¿Cuántos puntos?: 1000 Pi estimado es: 3.2440</pre>	<pre>>>> ¿Cuántos puntos?: 5000 Pi estimado es: 3.1464</pre>
---	---