Introducción

Con el el siguiente trabajo se pretende determinar el numero irracional Pi mediante el uso del desarrollo en serie de Mac Laurin (serie de Taylor alrededor del cero) de la función "arctan". La serie puede expresarse de la siguiente manera:

$$\arctan(x) = \lim_{n\to\infty} P_n(x) = \lim_{n\to\infty} \sum_{k=1}^{n} (-1)^{k+1} \frac{x^{2k-1}}{(2k-1)}$$

La formula general del polinomio de Taylor es la siguiente:

$$f(x) = \sum \left(\frac{(f^{(k)}(a))}{k!}(x-a)^k\right) \quad \text{con} \quad 0<k<\infty$$

Como el desarrollo es alrededor del cero , $a=0$

Entonces,

$$f(x) = \sum \left(\frac{(f^{(k)}(0))}{k!}(x)^k\right) \quad \text{con} \quad 0<k<\infty$$

Para $k=0$, se tiene $\dfrac{(f^{(0)}(0))}{(0!)}(x)^0 = \dfrac{\arctan(0)}{1}(1) = 0$

Para $k=1$, se tiene $\dfrac{(f^{(1)}(0))}{(1!)}(x)^1 = \dfrac{((1+0)^{-1})}{1}(x) = x$

Para $k=2$, se tiene $\dfrac{(f^{(2)}(0))}{(2!)}(x)^2 = \dfrac{(0(1+0)^{-2})}{2}(x)^2 = 0$

Para $k=3$, se tiene $\dfrac{(f^{(3)}(0))}{(3!)}(x)^3 = \dfrac{(-2+0)(1+0)^{-4}}{6}(x)^3 = \dfrac{-x^3}{3}$

Para $k=4$, se tiene $\dfrac{(f^{(5)}(0))}{(5!)}(x)^5 = \dfrac{(0)(1+0)^{-8}}{24}(x)^4 = 0$

Para $k=5$, se tiene $\frac{(f^{(5)}(0))}{(5!)}(x)^5=\frac{(24)(1+0)^{-16}}{120}(x)^5=\frac{x^5}{5}$

Para $k=6$, se tiene $\frac{(f^{(6)}(0))}{(6!)}(x)^6=\frac{0(1+0)^{-32}}{720}(x)^6=0$

Para $k=7$, se tiene $\frac{(f^{(7)}(0))}{(7!)}(x)^7=\frac{(-720)(1+0)^{-64}}{5040}(x)^7=\frac{-x^7}{7}$

Si sumamos cada uno de los términos obtenidos, tenemos:

$$\arctan(x)=x-\frac{x^3}{3}+\frac{x^5}{5}-\frac{x^7}{7}+K$$

donde K representa la suma de los términos para $k>7$

Expresando como una sumatoria se tiene:

$$\arctan(x) = \lim_{n\to\infty} P_n(x) = \lim_{n\to\infty} \sum_{k=1}^{n} (-1)^{k+1} \frac{x^{2k-1}}{(2k-1)}$$

Análisis y Conclusiones

Para el cálculo estimado de Pi se utilizaron funciones en base al arcotangente, el cual aproximamos utilizando la serie de Maclaurin, truncándola de acuerdo a nuestras necesidades de precisión.

En el caso del punto A, utilizamos la función:

$$\tan(\frac{\pi}{4})=1$$

De la cual despejamos Pi:

$$\pi=4*\arctan(1)$$

Llegamos a la precisión requerida (½ . 10^{-4}) en 20.000 pasos, y el valor estimado de Pi fue 3,1415.

Mientras tanto, en el punto B, la ecuación que utilizamos fue:

$$\frac{\pi}{4}=\arctan(\frac{1}{2})+\arctan(\frac{1}{3})$$

Y la conclusión que obtuvimos, tras solamente 7 iteraciones, fue que, a pesar de acercarse rápidamente al resultado, se estabiliza a una distancia del valor real, mayor a la precisión requerida. Por lo tanto, decidimos truncar ahí el método, ya que seguirlo no hubiese producido una mejora sustancial en el resultado. Obteniendo un valor aproximado de Pi = 3,1403.

Por último, en el punto C, se utilizó la fórmula de Machin:

$$\frac{\pi}{4}=4*\arctan\left(\frac{1}{5}\right)-\arctan\left(\frac{1}{239}\right)$$

Y en este caso, alcanzamos una distancia de ½ . 10^{-4} con respecto al valor de Pi proveído por la computadora, en tan solo 3 pasos, estimando Pi a 3,1416.

Como conclusión definitiva, vemos que en el primer caso, si bien se obtiene el resultado esperado, la cantidad de iteraciones requeridas es muy grande, cosa que no sucede en el tercer caso, donde la precisión se alcanza muy rápido.
Y en el segundo caso, tenemos la ventaja de que se acerca velozmente al valor buscado, pero una vez cerca, se estabiliza quedando alejado de Pi en un orden mayor al de la precisión requerida.

En todos estos casos, el error que obtenemos es debido a que truncamos una serie infinita.
Además, al realizar cada operación, redondeamos los valores obtenidos, a cuatro decimales significativos, lo que acarrea un error de redondeo.
Este último error, se puede mejorar, ampliando la grilla, para utilizar más decimales significativos.
Mientras que el error de truncamiento, no se puede evitar ya que, por obvias razones, una suma de infinitos términos no se puede realizar. Aunque sí podríamos mejorarlo, utilizando más iteraciones.

Desarrollo

Tp1.java

```java
import java.text.*;

public class tp1 {


        private static final double PRECISION_A = 0.00005; /* Precisión 0.5 * 10^(-4) */
        private static final double PRECISION_B = PRECISION_A;
        private static final double PRECISION_C = PRECISION_A;
        private static final int CANTIDAD_DE_DECIMALES = 4;
        private static final int NUM_MAX_ITERACIONES = 100000;

        public static void main (String[] args) {

            int numeroDeIteracion;
            double precision;
                double precisionAnterior = 1;
                double arcotangente = 0;
                double arcotangente1 = 0;
                double arcotangente2 = 0;
                boolean precisionEstabilizada = false;
                Arctan arctan1;
                Arctan arctan2;

            /* PUNTO A */
                System.out.println ("Punto A\n");
                arctan1 = new Arctan(CANTIDAD_DE_DECIMALES);

            numeroDeIteracion = 1;
            precision=1;
            arcotangente = 0;
            precisionEstabilizada = false;
```

```java
		while ( (precision > PRECISION_A) && ( numeroDeIteracion <= NUM_MAX_ITERACIONES ) && 
( !precisionEstabilizada ) ){

			arcotangente = arctan1.calcular(numeroDeIteracion,1);
			precision = Math.abs(4*arcotangente - Math.PI);
			imprimirPrecision (precision,numeroDeIteracion);
			if (Math.abs(precisionAnterior - precision) < PRECISION_B) 
precisionEstabilizada = true;
			numeroDeIteracion++;
		}

		imprimirPiEstimado (4*arcotangente);
		imprimirResultado(precision,PRECISION_A,numeroDeIteracion-1,precisionEstabilizada);

		/* FIN PUNTO A */

		/* PUNTO B */

		System.out.println ("\n\nPunto B\n");

		arctan1 = new Arctan(CANTIDAD_DE_DECIMALES);
		arctan2 = new Arctan(CANTIDAD_DE_DECIMALES);

		precision = 1;
		numeroDeIteracion = 1;
		precisionAnterior = 1;
		arcotangente1 = 0;
		arcotangente2 = 0;
		precisionEstabilizada = false;

		while ( (precision > PRECISION_B) && ( numeroDeIteracion <= NUM_MAX_ITERACIONES ) && 
( !precisionEstabilizada ) ){
			arcotangente1 = arctan1.calcular(numeroDeIteracion,0.5);
			arcotangente2 = arctan2.calcular(numeroDeIteracion, 0.333);
			precisionAnterior = precision;
			precision = Math.abs(4*(arcotangente1 + arcotangente2) - Math.PI);
			imprimirPrecision (precision,numeroDeIteracion);
			if (Math.abs(precisionAnterior - precision) < PRECISION_B) 
precisionEstabilizada = true;
			numeroDeIteracion++;

		}

		imprimirPiEstimado(4*(arcotangente1 + arcotangente2));
		imprimirResultado(precision,PRECISION_B,numeroDeIteracion-1,precisionEstabilizada);

		/* FIN PUNTO B */


		/* PUNTO C */
		System.out.println ("\n\nPunto C\n");

		arctan1 = new Arctan(CANTIDAD_DE_DECIMALES);
		arctan2 = new Arctan(CANTIDAD_DE_DECIMALES);

		precision = 1;
		numeroDeIteracion = 1;
		precisionAnterior = 1;
		arcotangente1 = 0;
		arcotangente2 = 0;
		precisionEstabilizada = false;

		while ( (precision > PRECISION_C) && ( numeroDeIteracion <= NUM_MAX_ITERACIONES ) && 
( !precisionEstabilizada ) ){
			arcotangente1 = arctan1.calcular(numeroDeIteracion,0.2);
			arcotangente2 = arctan2.calcular(numeroDeIteracion, 0.004184);
			precisionAnterior = precision;
			precision = Math.abs(4*(4*arcotangente1 - arcotangente2) - Math.PI);
			imprimirPrecision (precision,numeroDeIteracion);
			if (Math.abs(precisionAnterior - precision) < PRECISION_C) 
precisionEstabilizada = true;
			numeroDeIteracion++;
		}

		imprimirPiEstimado(4*(4*arcotangente1 - arcotangente2));
		imprimirResultado(precision,PRECISION_C,numeroDeIteracion-1,precisionEstabilizada);

		/* FIN PUNTO C */
	}
```

```java
	private static void imprimirPiEstimado(double valor) {
		DecimalFormat numeroFormateado; /* Objeto para darle formato a los números */
		numeroFormateado = new DecimalFormat("0.00000");
		System.out.println("El valor de PI estimado es: "+numeroFormateado.format(valor));
	}

	private static void imprimirResultado(double precision, double precisionRequerida, int
numeroDeIteracion, boolean precisionEstabilizada) {

		if (precision < precisionRequerida) System.out.println ("Se alcanzó la precisión en
la iteración "+numeroDeIteracion);
		else
			if (numeroDeIteracion == NUM_MAX_ITERACIONES) System.out.println ("Se superó
el máximo de "+NUM_MAX_ITERACIONES+" iteraciones.");
			else
				if (precisionEstabilizada) System.out.println("La precisión no varió
entre las iteraciones "+(numeroDeIteracion - 1)+" y "+numeroDeIteracion);
	}

	private static void imprimirPrecision(double precision,    int numeroDeIteracion) {

		DecimalFormat numeroFormateado; /* Objeto para darle formato a los números */
		numeroFormateado = new DecimalFormat("0.00000");

		if (numeroDeIteracion == 1) System.out.println ("Precisión\tIteración");

		if (		( numeroDeIteracion <= 20 )
				||  ( ((numeroDeIteracion % 50) == 0) && (numeroDeIteracion <= 1000) )
				||  ( ((numeroDeIteracion % 1000) == 0) && (numeroDeIteracion >
1000) )
			)

			System.out.println(numeroFormateado.format(precision)
+"\t\t"+numeroDeIteracion);
	}

}
```

Arctan.java

```java
public class Arctan {

	double resultado;
	int cantidadDeDecimales;

	public Arctan(int cantidadDeDecimales){
		resultado = 0;
		this.cantidadDeDecimales = cantidadDeDecimales;
	}

	public double calcular(int i, double x){
		resultado += ( redondear(Math.pow(-1,i+1),cantidadDeDecimales)*redondear
(Math.pow(x,2*i - 1),cantidadDeDecimales )) / ( 2*i - 1 );
		return resultado;
	}

  public double redondear (double x,int cantidadDeDecimales)
    {
	int redondeo = (int) Math.pow(10,cantidadDeDecimales+1);
	long y = Math.round (x*redondeo);
	return (double) y /redondeo;
    }


}
```

Impresión de resultados

Punto A

Precisión	Iteración

0,85841	1
0,47493	2
0,32507	3
0,24635	4
0,19809	5
0,16555	6
0,14215	7
0,12452	8
0,11077	9
0,09975	10
0,09072	11
0,08319	12
0,07681	13
0,07134	14
0,06659	15
0,06244	16
0,05877	17
0,05551	18
0,05260	19
0,04997	20
0,02000	50
0,01000	100
0,00667	150
0,00500	200
0,00400	250
0,00333	300
0,00286	350
0,00250	400
0,00222	450
0,00200	500
0,00182	550
0,00167	600
0,00154	650
0,00143	700
0,00133	750
0,00125	800
0,00118	850
0,00111	900
0,00105	950
0,00100	1000
0,00050	2000
0,00033	3000
0,00025	4000
0,00020	5000
0,00017	6000
0,00014	7000
0,00012	8000
0,00011	9000
0,00010	10000
0,00009	11000
0,00008	12000
0,00008	13000
0,00007	14000
0,00007	15000
0,00006	16000
0,00006	17000
0,00006	18000
0,00005	19000
0,00005	20000

El valor de PI estimado es: 3,14154
Se alcanzó la precisión en la iteración 20000

Punto B

```
Precisión   Iteración
0,19041          1
0,02550          2
0,00277          3
0,00195          4
0,00106          5
0,00124          6
0,00120          7
El valor de PI estimado es: 3,14039
La precisión no varió entre las iteraciones 6 y 7
```

Punto C

```
Precisión   Iteración
0,04169          1
0,00098          2
0,00004          3
El valor de PI estimado es: 3,14164
Se alcanzó la precisión en la iteración 3
```