



UNIVERSIDAD POLITÉCNICA DE TLAXCALA

REGIÓN PONIENTE

MATERIA:

Física

NOMBRE DEL ASESOR:

Ing. Vanesa Tenopala Zavala

NOMBRE DEL ALUMNO:

Pérez Pérez Denise

Moreno Bonilla Aldo

Garcia Flores Luis David

LICENCIATURA:

Ingeniería en Sistemas Computacionales

NOMBRE DE LA ACTIVIDAD:

Examen Primer Parcial – Cifras Significativas

FECHA:

25 de enero de 2024

Contenido

| | |
|--|----------|
| Introducción | 3 |
| Desarrollo | 4 |
| <i>Cifras significativas.</i> | <i>4</i> |
| <i>Reglas para determinar cuántas cifras significativas existen en el resultado.</i> | |
| | 4 |
| Regla 1. Los dígitos diferentes de cero son siempre significativos. | 4 |
| Regla 2. Cualesquiera ceros entre dos cifras significativas son | |
| significativos..... | 4 |
| Regla 3. Los ceros al final de la parte decimal son significativos. | 5 |
| Unos ejemplos más sencillos para entender estas 3 reglas son: | 5 |
| Evidencias de clase..... | 6 |
| Explicación de código:..... | 7 |
| <i>Importación de la clase JOptionPane:.....</i> | <i>7</i> |
| <i>Declaración de la clase principal:.....</i> | <i>7</i> |
| <i>Método principal main:.....</i> | <i>7</i> |
| <i>Declaración de variables:.....</i> | <i>7</i> |
| <i>Bucle while para la ejecución continua:</i> | <i>7</i> |
| <i>Entrada de usuario usando JOptionPane:</i> | <i>8</i> |
| <i>Llamada a la función contarCifrasSignificativas:.....</i> | <i>8</i> |
| <i>Cuadro de diálogo mostrando el resultado:</i> | <i>8</i> |
| <i>Cuadro de confirmación para continuar:</i> | <i>8</i> |

| | |
|---|-----------|
| <i>Clase interna ResultadoCifras:</i> | 9 |
| <i>Función contarCifrasSignificativas:</i> | 9 |
| <i>Eliminación de caracteres no deseados:</i> | 9 |
| <i>Bucle para contar cifras significativas:</i> | 10 |
| <i>Construcción del objeto ResultadoCifras:</i> | 10 |
| <i>Ejecución:</i> | 11 |
| <i>Fase 1:</i> | 11 |
| <i>Fase 1.1:</i> | 11 |
| <i>Fase 2:</i> | 12 |
| <i>Fase 2.2:</i> | 12 |
| <i>Fase 3:</i> | 13 |
| <i>Conclusión</i> | 14 |
| <i>Referencias</i> | 14 |

Introducción

Este informe presenta un programa en Java diseñado para abordar la manipulación de cifras significativas. El desarrollo de esta herramienta informática se centra en la comprensión y aplicación de las reglas fundamentales asociadas con las cifras significativas, ofreciendo a los usuarios una solución eficaz y fácil de usar. A lo largo de este informe, se examina el código fuente del programa, destacando sus características clave y proporcionando ejemplos ilustrativos para demostrar su funcionalidad.

Desarrollo

Cifras significativas.

Las cifras significativas aportan información sobre el resultado de alguna medición, y estas representan el uso de una o más escalas de incertidumbre en determinadas aproximaciones; como por ejemplo:

- Se dice que 4,7 tiene dos cifras significativas mientras que 4,07 tiene tres cifras significativas.

Con forme a esto se deberá realizar un programa en lenguaje de programación en JAVA, el cual deberá proporcionar la cantidad de cifras significativas que contiene cierto número proporcionado por un usuario, ya sea en decimal o en notación científica; y el cual deberá contener las siguientes 3 reglas para completar su función.

Reglas para determinar cuántas cifras significativas existen en el resultado.

Regla 1. Los dígitos diferentes de cero son siempre significativos.

Es decir, que un número como el 26.38 tendrá cuatro cifras significativas y un número como el 7.94 tendrá 3 cifras significativas. Ahora bien, los numero a la izquierda del número distinto de cero no son significativos.

Ejemplos:

- 0,03 (este número contiene una sola cifra significativa)
- 0,0000000000000000395 (este número contiene tres cifras significativas).

Regla 2. Cualesquiera ceros entre dos cifras significativas son significativos.

Supongamos que tenemos como resultado la medición el número 406. La regla 1 dice que el 3 y el 6 son significativos, pero ¿Qué es lo que sucede con el cero el cual forma parte de las decenas de este número?, a estos ceros se les llama

“ceros apretados” y estos deben de tenerse en cuenta, ya que los ceros en medio de números diferentes a cero son significativos.

Ejemplos:

- 902 (este número contiene tres cifras significativas).
- 10,609 (este número contiene cinco cifras significativas).

Regla 3. Los ceros al final de la parte decimal son significativos.

Esto quiere decir que los ceros que se encuentren del lado derecho de un número diferente a cero son significativos.

Algunos ejemplos de esta regla son los ceros significativos de color rojo:

- 0,00500.
- 0,03040.

Otros ejemplos de esta regla en notación científica son:

- 2.30x10⁻⁷.
- 4.500x10⁻¹².

Unos ejemplos más sencillos para entender estas 3 reglas son:

0,00003400.

0,0250600.

3.20100x10⁻⁷.

- Los ceros no son significativos después de un decimal antes de que no sean números cero.
- Todos los números distintos de cero son significativos
- Los ceros después de los números distintos de cero son significativos
- Los ceros que estén entre dos cifras distintas de cero son significativos

Evidencias de clase

Cifras significativas

- Las **cifras significativas** aportan información sobre el resultado de medición.
- Elas representan el uso de una o más escalas de incertidumbre en determinadas aproximaciones.
 - Por ejemplo, se dice que 4,7 tiene dos cifras significativas, mientras que 4,07 tiene tres.

Reglas para determinar cuántas cifras significativas existen en el resultado

- REGLA 1: LOS DÍGITOS DIFERENTES DE CERO SON SIEMPRE SIGNIFICATIVOS.**

Es así que un número como 26.38 tendría cuatro cifras significativas, mientras que 7.94 tendría tres.

- Ahora bien, los ceros a la izquierda del primer número distinto de cero no son significativos
 - > 0.03 (que tiene una sola cifra significativa)
 - > 0.0000000000000395 (este tiene solo tres)

REGLA 2: CUALESQUIERA CEROS ENTRE DOS CIFRAS SIGNIFICATIVAS SON SIGNIFICATIVOS.

- Supongamos que tienes como resultado de una medición el número 406.
- Según la **Regla 1**, el 4 y el 6 son significativos.
- Pero ¿Qué sucede con el cero que forma parte de las decenas en este número? A estos se les llama "ceros apesados" y deben ser tenidos en cuenta.
- Los ceros situados en medio de números diferentes de cero son significativos
 - > 901 cm (que tiene tres cifras significativas)
 - $> 10,609$ kg (teniendo cinco cifras significativas)

Regla 3: los ceros al final de la parte decimal son significativos.

- Veamos dos ejemplos de esta regla, con los ceros significativos en rojo

- 0.00500
- 0.03040

- Y aquí otros dos ejemplos con los ceros significativos en rojo:

- 2.30×10^{-5}
- 4.500×10^{12}

Cifras significativas

$0.00321 = 3.21 \times 10^{-3}$

3 cifras significativas

Cifras Significativas (c.s.) = Son los dígitos obtenidos de una medida.

Una propiedad importante de las c.s. es que el número de c.s. no dependen del punto decimal.

| | |
|--------|-----------------------------------|
| 255 | Aquí todos estos n° tienen 3 c.s. |
| 25,5 | |
| 2,55 | |
| 0,255 | |
| 0.0255 | |

Cifras Significativas

0,00003400

Los ceros después de números distintos de cero en un decimal son significativos

Todos los números distintos de cero son significativos

Los ceros no son significativos después de un decimal antes de que no sean números cero

Explicación de código:

Importación de la clase JOptionPane:

```
import javax.swing.JOptionPane;
```

Importa la clase JOptionPane del paquete javax.swing, que proporciona métodos para mostrar cuadros de diálogo en una interfaz gráfica de usuario.

Declaración de la clase principal:

```
public class CifrasSignificativas {
```

Define la clase principal llamada CifrasSignificativas.

Método principal main:

```
public static void main(String[] args) {
```

Inicia el método principal del programa. Aquí se realiza la lógica principal del programa.

Declaración de variables:

```
boolean continuar = true;
```

Declara una variable booleana continuar que se utiliza para controlar el bucle while y determinar si el usuario desea ingresar otro número.

Bucle while para la ejecución continua:

```
while (continuar) {
```

Inicia un bucle while que permite al usuario ingresar números continuamente hasta que decida no continuar.

Entrada de usuario usando JOptionPane:

```
String numeroIngresado = JOptionPane.showInputDialog("Ingrese un  
número en notación científica o número normal:");
```

Muestra un cuadro de diálogo para que el usuario ingrese un número en notación científica o formato normal y guarda la entrada en la variable numeroIngresado.

Llamada a la función contarCifrasSignificativas:

```
ResultadoCifras resultado = contarCifrasSignificativas(numeroIngresado);
```

Llama a la función contarCifrasSignificativas para contar las cifras significativas del número ingresado.

Cuadro de diálogo mostrando el resultado:

```
JOptionPane.showMessageDialog(null, "Número ingresado: " +  
numeroIngresado +  
"\nCifras significativas: " + resultado.cifrasSignificativas +  
"\nCifras encontradas: " + resultado.cifrasEncontradas);
```

Muestra un cuadro de diálogo con el número ingresado y el resultado del conteo de cifras significativas.

Cuadro de confirmación para continuar:

```
int respuesta = JOptionPane.showConfirmDialog(null, "¿Desea ingresar otro  
número?", "Continuar", JOptionPane.YES_NO_OPTION);  
continuar = (respuesta == JOptionPane.YES_OPTION);
```

Muestra un cuadro de confirmación para preguntar al usuario si desea ingresar otro número. La respuesta se utiliza para actualizar la variable continuar y determinar si el bucle while debe continuar.

Clase interna ResultadoCifras:

```
private static class ResultadoCifras {

    int cifrasSignificativas;

    String cifrasEncontradas;

    ResultadoCifras(int cifrasSignificativas, String cifrasEncontradas) {

        this.cifrasSignificativas = cifrasSignificativas;

        this.cifrasEncontradas = cifrasEncontradas;

    }

}
```

Define una clase interna llamada ResultadoCifras, que se utiliza para almacenar el resultado del conteo de cifras significativas.

Función contarCifrasSignificativas:

```
private static ResultadoCifras contarCifrasSignificativas(String numero) {
```

Define una función que toma un número como entrada y devuelve un objeto ResultadoCifras que contiene el conteo de cifras significativas y las cifras encontradas.

Eliminación de caracteres no deseados:

```
String numeroLimpio = numero.replaceAll("[^0-9\\.\eE-]", "");
```

Utiliza expresiones regulares para eliminar cualquier caracter no deseado del número, dejando solo dígitos, el punto decimal, la notación científica y el signo negativo.

Bucle para contar cifras significativas:

```
for (int i = 0; i < numeroLimpio.length(); i++) {  
    }  
}
```

Itera sobre cada carácter del número limpio y cuenta las cifras significativas según las reglas dadas.

Construcción del objeto ResultadoCifras:

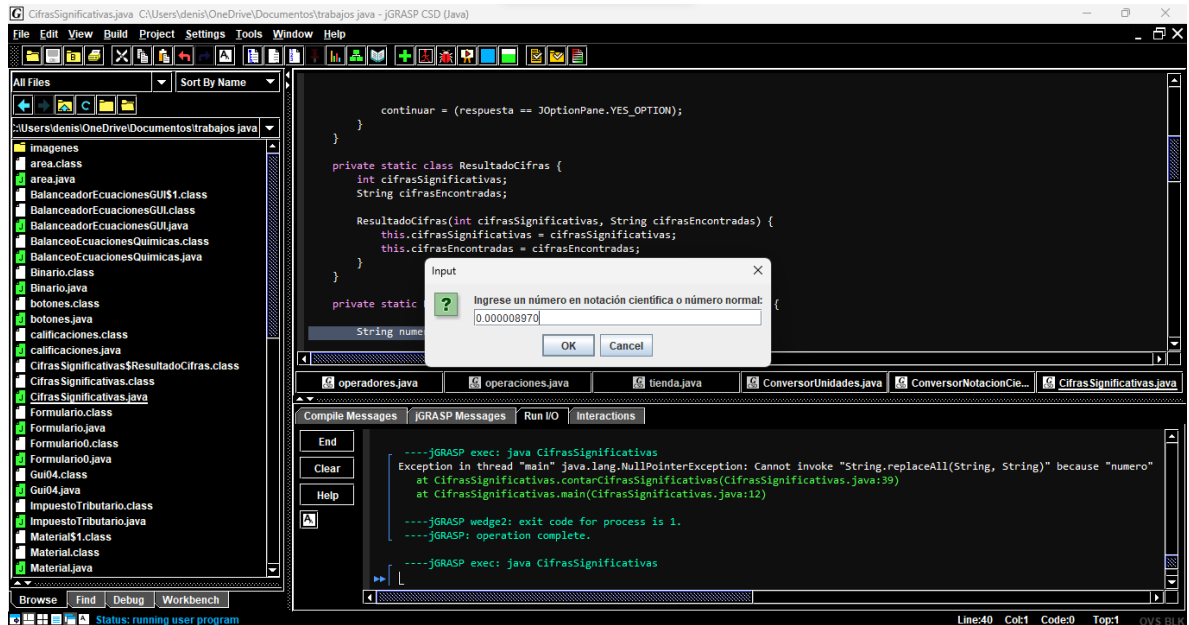
```
return new ResultadoCifras(conteo, cifrasEncontradas);
```

Devuelve un nuevo objeto ResultadoCifras con el conteo de cifras significativas y las cifras encontradas.

Ejecución:

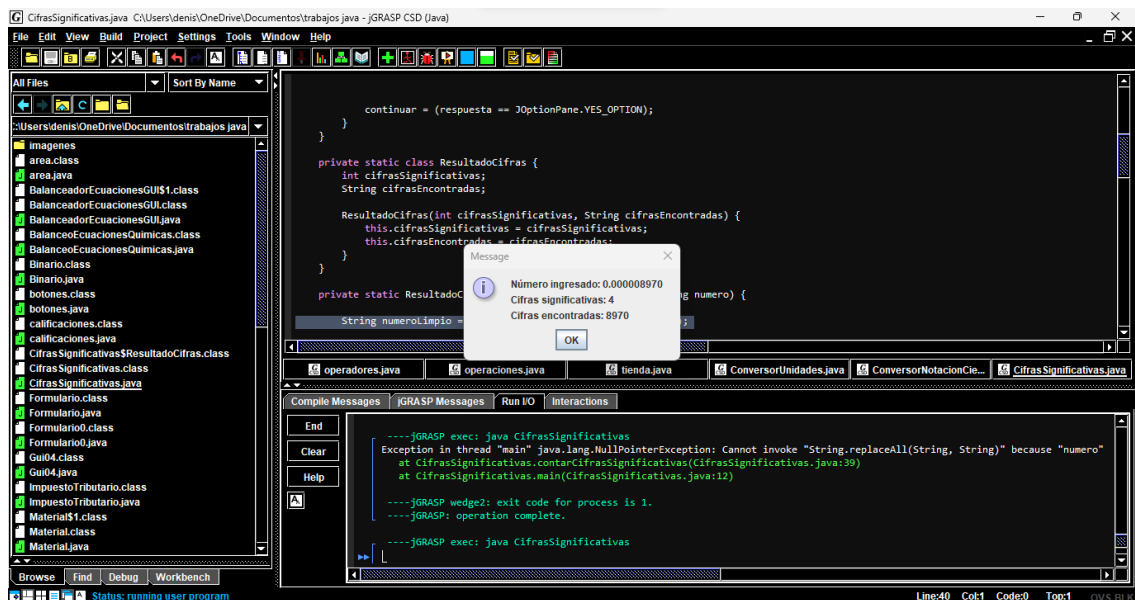
Fase 1:

Se solicita al usuario el número.



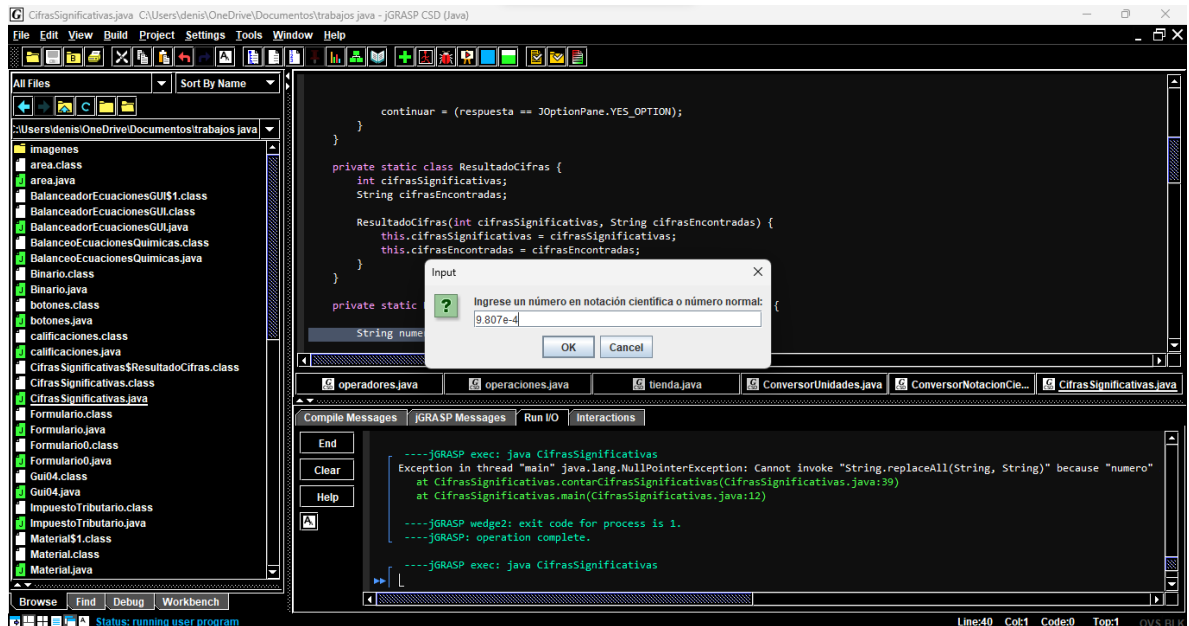
Fase 1.1:

El usuario podrá visualizar el número ingresado, la cantidad de cifras significativas y cuales son.

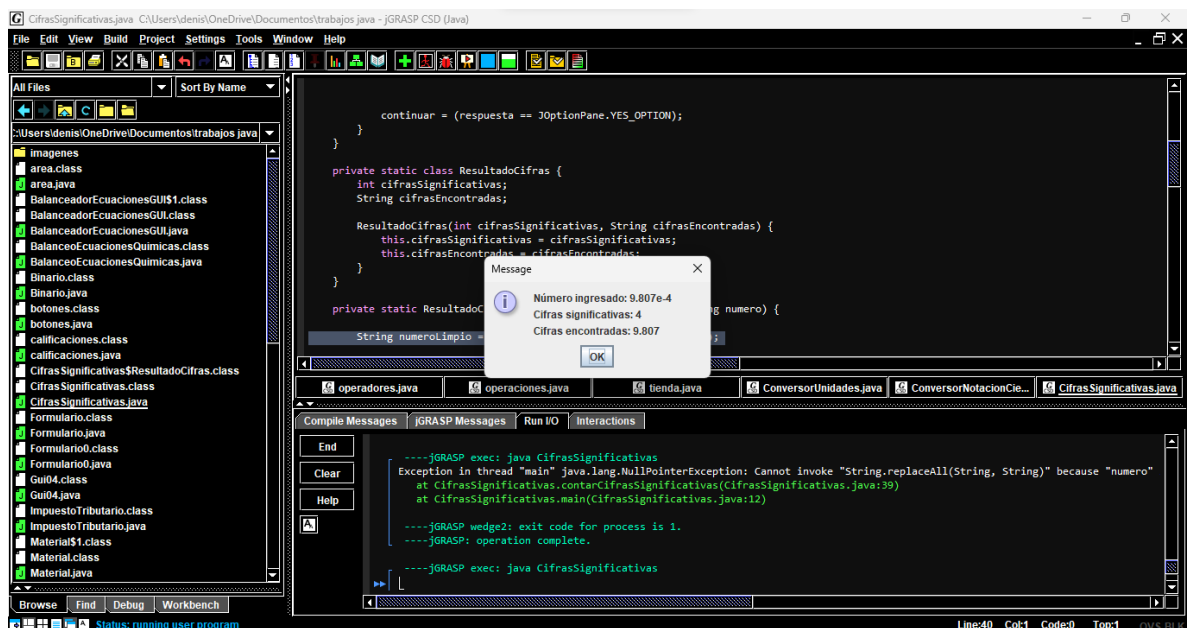


Fase 2:

El usuario ingresa la notación científica.

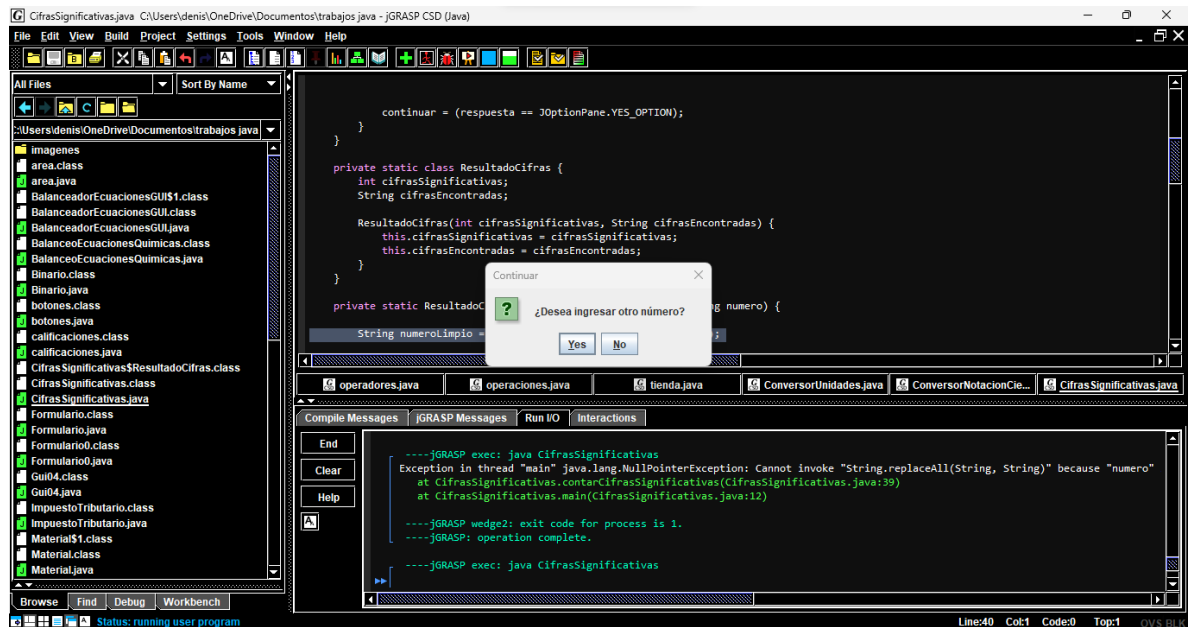
**Fase 2.2:**

El usuario podrá visualizar el número ingresado, la cantidad de cifras significativas y cuales son.



Fase 3:

Si el usuario desea ingresar otra cantidad, visualizará una ventana emergente donde podrá seleccionar si lo desea o no.



Al seleccionar si, podrá ingresar una nueva cantidad de lo contrario el programa termina.

Conclusión

Las cifras significativas representan las reglas que alguien utiliza al medir o realizar cálculos en ciencia. Estas reglas le permiten expresar cuánta certeza tiene en sus resultados. Al seguirlas, una persona puede comunicar de manera más clara y honesta la confiabilidad de sus mediciones. En síntesis, las cifras significativas funcionan como un conjunto de reglas o pautas que nos permiten utilizar para hablar con precisión acerca de resultados científicos o matemáticos, evitando exageraciones o subestimaciones.

Referencias

- Java Platform SE 8. (2024). <https://docs.oracle.com/javase/8/docs/api/>