

Trabajo Introducción a la Informática y a la programación curso 2021-2022

Se pretende implementar una calculadora sencilla. Las operaciones a realizar son: multiplicación, división, resta y suma.

A continuación, se indican ejemplos de resolución.

Multiplicación	División	Resta	Suma
>> -2*-2 >> 4	>> -12.5/-2 >> 6.25	>> -12.5-3 >> -15.5	>> 12.5+3.8 >> 16.3
>> (-2)*(-2) >> 4	>> (-2)/(+(-5)) >> 0.4	>> (+5)-(-3) >> 8	>> (-2.1)+(-0.9) >> -3

Inicio:

Nombre del programa: Calculadora.java

Consideración inicial:

El programa estará activo e irá permitiendo la introducción de operaciones tras resolver la operación hasta que se escriba por teclado la palabra “salir”. Use para ello un bucle do{ ...}while();

Creación teclado y línea entrada datos

Cree el código para la lectura de los datos usando el Scanner.

A continuación, añada:

```
System.out.print(">> "); //Espera lectura de la fórmula
```

Entrada dato:

String *formula*

El dato de entrada será del tipo String y contendrá dos números y la operación a realizar.

En la expresión podrán aparecer paréntesis.

Observaciones, con la ayuda de la función replace(), haga que.

1. El sistema sea robusto a espacios en blanco introducidos. 2+3 ó 2 + 3
2. Sea robusto a la existencia o no de paréntesis.
3. Sea capaz de resolver operaciones del tipo ++, --, +-, -+, es decir, sabrá que +-5 es igual a -5.

De tal forma que la expresión leída como String, tras estos pasos ha de quedar, por ejemplo.

$(+ 2) - (- 3) \rightarrow +2+3$

$(-(-5))^* (-4) \rightarrow +5*-4$

Selección de la operación a realizar y de la posición del operador en la cadena

Tras la ejecución de este bloque, se tendrán actualizadas dos variables. Una cuál es la operación a realizar, y otra el lugar donde se encuentra el operador (*,/,+,-).

```
String op="";  
int ind=0;
```

```
//Implemente el bloque de selección para actualizar op e ind.
```

```
//Consideraciones:
```

- Pregunte por si está o no el signo *, /, - y +-.
- No debe considerar como signo de la operación los signos que indican que el número es positivo o negativo.

Para ello use una estructura if, else if, else.

En el bloque else, de dicha estructura, indicaremos el caso de que el operador no sea ninguno de los indicados.

```
}else{  
    System.out.println("Operación no válida");  
    System.exit(0); //Abandona ejecución.  
}
```

Extracción de los elementos de la operación y conversión a double.

Creamos las variables,

```
double a=0.0;  
double b=0.0;
```

Extraiga los valores de a y b del String formula. Use el valor de ind obtenido en el bloque de selección. Use el método de la clase envolvente Double, parseDouble(), para obtener los elementos de la operación a y b.

Tras la ejecución de este bloque tendremos los valores con los que realizar la operación nos indique la variable String `op`;

Por ejemplo `+5.24+-2.3`, tendremos que `a=5.24` y `b=-2.3`. Recuerda que `Double.parseDouble("+5.24")`, es 5.4

Bloque realizamos la operación.

Cree la variable result del tipo double donde almacenar el resultado de la operación. Con la ayuda de la estructura de selección switch(), evalúe result en cada caso.

```
switch(op){  
    case "+": result=  
    case "-": result=  
    case "*": result=  
    case "/": result=  
}
```

Imprime resultado consola.

Si se observa en la tabla del principio, si el resultado de la operación presente parte decimal cero, el formato de la impresión es entero. Es decir, no muestra -3.0, sino que muestra 3.

Para ello implemente un código que si la parte decimal es 0.0 imprima result como entero.

```
System.out.print(">> ")
```

Por último, si ha usado, la clase Scanner, cierre el teclado.

Mejoras:

- 1) Inserte una estructura para capturar la excepción que podemos tener al realizar la conversión de String a double en el código implementado.

```
try{  
    //Código conversión String->Double  
}catch(Exception e){  
    System.out.println("Fórmula errónea");  
}
```

- 2) Agrupe diferentes partes del código en métodos (según su consideración). Retoque el método principal para que realice la ejecución con las llamadas a los métodos haya realizado.
- 3) Incluya otras operaciones, tales como elevar un número a otro, etc.