



INTRODUCCIÓN A LAS CIENCIAS COMPUTACIÓN

TAREA 6

Fecha: 15 de Octubre de 2016

Problemas mientras se toma el café

Problema 1: Escriba un programa que imprima una caja, un ovalo, una flecha y un diamante. La salida de tu programa debe de parecerse a esto:

```
*****      ***      *      *
*      *      *      *      ***      *      *
*      *      *      *      *****      *      *
*      *      *      *      *      *      *
*      *      *      *      *      *      *
*      *      *      *      *      *      *
*      *      *      *      *      *      *
*      *      *      *      *      *      *
*****      ***      *      *
```

Problema 2: Escriba un programa que introduzca tres diferentes enteros, después que imprima la suma, el promedio, el producto, el número más pequeño y el más grande de ellos. Utilice la estructura de control *if*. La salida de tu programa debe aparecer de la siguiente manera:

```
Escriba tres enteros diferentes:
13
27
14
El promedio es 54
El producto es 18
El número más pequeño es 13
El número más grande es 27
```

Problema 3: Escriba un programa que introduzca un número de 6 cifras, que lo separe en sus dígitos individuales y que los imprima en orden inverso separados por un tabulador. NOTA: Un tabulador es un carácter especial que equivale aproximadamente a 5 espacios y se representa por `\t`

```
Introduce un número de 6 cifras:
642139
9      3      1      2      4      6
```

Problema 4: El interés simple para un préstamo se calcula mediante la fórmula:

$$\text{Interés} = \text{préstamo} * \text{tasa} * \text{días} / 365$$

La fórmula anterior asume que tasa es la tasa de interés anual, y por lo tanto incluye la división entre 365 (días). Desarrolla un programa que introduzca préstamo, tasa y días para varios préstamos, y que calcule y despliegue el interés simple para cada préstamo, utilizando la fórmula anterior. A continuación se muestra un ejemplo del diálogo de entrada/salida:

```
Introduzca el monto del préstamo (-1 para terminar): 1000.00
Introduzca la tasa de interés: 0.1
Introduzca el periodo del préstamo en días: 365
El monto del interés es $100.00
```

```
Introduzca el monto del préstamo (-1 para terminar): 1000.00
Introduzca la tasa de interés: 0.08375
Introduzca el periodo del préstamo en días: 224
El monto del interés es $51.40
```

```
Introduzca el monto del préstamo (-1 para terminar): 1000.00
Introduzca la tasa de interés: 0.09
Introduzca el periodo del préstamo en días: 1460
El monto del interés es $3600.00
```

```
Introduzca el monto del préstamo (-1 para terminar): -1
```

Problema 5: Escribe un programa que dado un número despliegue el perímetro de un cuadrado de tamaño nxn. Por ejemplo, si su programa lee un tamaño 5, debe desplegar:

```
*****
*   *
*   *
*   *
*****
```

Problema 6: Supón que tienes la siguiente función:

$$f(x) = \begin{cases} x+1 & \text{si } x=3 \\ 2x & \text{si } x=6 \\ x*x & \text{si } x=-4 \\ -2x*x+x-3 & \text{si } x=0 \\ 0 & \text{en otro caso} \end{cases}$$

Evalúala de -10 a 10 con un ciclo *for*. Utiliza un *switch* para representar la función.

```
x=-10    f(x)=0
...
x=-4     f(x)=16
...
x=10     f(x)=0
```

Problema 7: Un algoritmo para calcular el mínimo común múltiplo de dos número puede ser el siguiente: Sea a , b los números de los que se quiera obtener el mínimo común múltiplo, establezca las variables aux , y $count$ igual a 1. Hacer $aux=count*a$; e incrementar $count$ en una unidad mientras aux modulo b sea distinto de cero. Luego aux será el mínimo común múltiplo entre a y b .

Implementa este algoritmo y realiza una prueba con $a=237$ y $b=53$ (Tip: Utiliza un *do while*)

Ingresa el valor para a : 237

Ingresa el valor para b : 53

El mcd de (237,53) es: 3

Problema 8: Dado un numero n imprimir una piramide invertida, por ejemplo, si su programa lee un tamaño 5, debe desplegar. TIP: Utiliza funciones.

```
*****
```

```
****
```

```
***
```

```
**
```

```
*
```

Representación de fracciones.

Las fracciones forman parte de la educación matemática básica. Desde la primaria hemos aprendido y manipulado estos objetos hasta llegar a dominar las operaciones entre ellos. Una fracción se representa mediante un numerador y un denominador.

Implementación

Completar la clase **Fraccion.java** la cual debe implementar la representación de fracciones antes mencionada. La clase Fracción consta de las siguientes operaciones:

1. Los métodos sin modificación entre fracciones: *suma*, *resta*, *multiplicación* y *división*. Estas operaciones tienen la forma *Fraccion operation(Fraccion operator);*.
2. Implementa las mismas operaciones anteriores pero ahora utilizando métodos con modificación.
3. Operaciones de comparación: *menor que*, *mayor que*, *igual*. Estas operaciones tienen la forma *Fraccion comparisson(Fraccion operator);*. Estas operaciones devuelven un tipo boolean.
4. Operaciones de conversión: conversión a cadena, conversión a punto flotante de precisión simple. Estas operaciones devuelven un objeto del tipo solicitado en la conversión, *String toString()*; para cadena y *float getFloatValue()*; para punto flotante.

La clase Fraccion tiene tres constructores con la siguiente forma:

1. *public Fraccion(int n, int d)*. Este constructor toma el numerador y el denominador y construye una fracción con ellos.
2. *public Fraccion(int n)*. Este constructor toma el numerador y el denominador lo establece en 1.
3. *public Fraccion(String s)*. Este constructor toma una cadena de la forma "num/den" y la convierte a una fracción.

Además, la clase Fraccion debe tener métodos para devolver el numerador y el denominador de la fracción que los invoque. Finalmente, cada que se convierta una fracción a otro tipo de dato (cadena o número de punto flotante) se debe regresar la forma más simplificada.

Por ejemplo, la invocación `new Fraccion(6,4).toString()` debe devolver la cadena "3/2", no la original que sería "6/4".

Otra cosa que hay que notar en cuanto a la abstracción de las fracciones, es que el denominador

no puede ser cero, por lo tanto si algun constructor u operación cae en este caso se tiene que establecer la fracción "104723/104729"

Consejos de implementación.

- El algoritmo de Euclides.
Para la simplificación de una fracción basta encontrar el máximo común divisor (mcd) y dividir tanto el numerador como el denominador entre este número.
- Simplificación.
Dado que ninguna operación modifica al objeto que la invoca, se debe considerar simplificar la fracción dentro del constructor, con esto se logran dos cosas, primero que las fracciones sean almacenadas de manera simplificada y que cuando se cree un objeto Fraccion como resultado de una operación, este objeto ya esté simplificado también.

Programa de Prueba.

Se debe crear una clase de prueba **Main.java** en donde solo se alojará la función main que probará el funcionamiento de nuestro programa. Con comentarios define al menos 2 casos de prueba para cada operación, seguido del código y verifica que tu programa funcione de manera correcta.

Un ejemplo de tu clase Main sería:

```
public class Main{  
    public static void main(String[] args){  
        //PRUEBA 1. Crear e imprimir la fracción 1/1  
        Fraccion f = new Fraccion("1/1");  
        System.out.println(f);  
    }  
}
```