

Hoja de trabajo No. 2

Realizar: Programa para evaluar expresiones postfix.

Realizarse: en forma **INDIVIDUAL**.

Fecha de entrega: martes 10 de agosto

Objetivos:

- Utilización de genéricos.
- Diseño del ADT para pilas (stack).
- Implementación de pilas con un vector de tamaño variable.
- Control de versiones del programa.
- Emplear JUnit para casos de prueba.

Programa a realizar:

Su programa debe leer, de la manera que mejor le convenga a usted, una expresión en formato Postfix y producir el resultado de la misma.

Adjunto un ejemplo tomado de la wikipedia:

Por ejemplo, el cálculo: $((1 + 2) * 4) + 3$, puede ser anotado como en notación postfija con la ventaja de no prevalecer las normas y los paréntesis necesarios:

1 2 + 4 * 3 +

La expresión es evaluada de izquierda a derecha utilizando una pila:

Apilar cuando se enfrentan a un operando y

Desafilar dos operandos y evaluar el valor cuando se enfrentan a una operación.

Apilar el resultado.

De la siguiente manera (la Pila se muestra después de que la operación se haya llevado a cabo):

ENTRADA	OPERACION	PILA
1	Apilar operando	1
2	Apilar operando	1, 2
+	Sumar	3
4	Apilar operando	3, 4
*	Multiplicar	12
3	Apilar operando	12, 3
+	Añadir	15

Tareas:

- Diagrama UML de clases y de secuencia. NO utilice ingeniería reversa. En el diagrama de secuencia modele la comunicación entre objetos para lograr la evaluación de la expresión Postfix. Use notación UML 2
- Construir la interfaz de la Pila, su clase abstracta y la clase de implementación con un vector de tamaño variable. Debe utilizar genéricos.
- Su programa principal debe permitir cambiar la clase pila, sin que se vea afectado en su funcionamiento.
Nota: En la siguiente hoja de trabajo se usará este mismo programa, pero con una implementación diferente para la pila.
- Debe dejar evidencia de todo el desarrollo en el repositorio de Subversion. Indicar como acceder a su repositorio y si es necesario, agregar a su catedrático y auxiliar para que tengan acceso al mismo.
- Incluya pruebas unitarias con JUnit para la clase pila.

Debe subir a Sakai todos los productos elaborados en los incisos a, b, c, e y los enlaces a su repositorio de Subversion.



Calificación: deben existir los diagramas de Clases y Secuencia para que sea calificado su programa.

Aspecto	Puntos
Estilo de codificación: comentarios, indentación, nombres de variables significativas.	5
Documentación generada con Javadoc, tiene precondiciones y postcondiciones en los métodos del ADT	5
Uso del repositorio: existen más de tres versiones guardadas, la última versión es igual a la colocada en el SAKAI	10
Diagrama de clases: muestran la abstracción y encapsulación de las operaciones.	5
Diagrama de secuencia muestra la evaluación de la expresión postfix	10
ADT pila, con uso de genéricos.	30
Funcionamiento del programa	25
Pruebas JUnit para las operaciones de la pila.	10
TOTAL:	100