Estructuras de Datos 2021-1
Práctica de Reposición: Pilas.

M. en C. Carlos Zerón Martínez
zeronmc@gmail.com

Emmanuel Cruz Hernández
emmanuel_cruzh@ciencias.unam.mx

José Antonio Vilchis Salazar
grand_paladin@ciencias.unam.mx

Sara Doris Montes Incin
isara22@ciencias.unam.mx

Fecha de entrega: 28 de enero de 2021
Hora de entrega: 23:59 hrs

1. Introducción

Se debe implementar un programa que permita almacenar cadenas de caracteres en una pila, dada cadena tiene a lo más 255 caracteres.

Cada localidad de memoria de la pila puede contener alguno de los siguientes elementos:

- Un entero que representa la longitud de la cadena que le antecede en la pila.
- Un caracter que forma parte de una de las cadenas que se almacenan en la pila.

Si tenemos la siguiente representación de una pila:

10
'R'
'e'
'p'
'o'
's'
'i'
'c'
'i'
'o'
'n'

El elemento que se encuentra en el tope es un *10*, que corresponde a la cantidad de caracteres que conforman la cadena próxima almacenada en la pila. El resto de los caracteres almacenados corresponde a un único caracter de una cadena.

2. Actividad

Deberán implementarse las siguientes operaciones:

2.1. `pushString(String)` *(2 puntos)*

Esta operación debe almacenar una cadena de longitud k en la pila. De acuerdo a la construcción anterior, se requerirá insertar k caracteres en la pila y luego insertar el entero k para tener referencia de la longitud de dicha cadena. Así, el último elemento insertado en la pila sería el número k.

2.2. `topString()` *(2 puntos)*

Nos devuelve un número k que a su vez nos indica que la primer cadena que podemos sacar de la pila es de tamaño k y se requerirían k operaciones `pop` para extraerla. Si por alguna razón no fuese posible almacenar dicha cadena, debe arrojarse una excepción.

2.3. `popString()` *(2 puntos)*

Esta operación debe devolver la última cadena insertada en la pila; conforme a la construcción anterior. Además, la cadena y su tamaño deben ser eliminados de la pila.

2.4. Menú *(2 puntos)*

El programa debe contar también con un menú que permita agregar y quitar cadenas de la pila, además de mostrar las salidas.

2.5. Escritura en archivos *(2 puntos)*

Las cadenas a insertar en la pila deben almacenarse en un archivo XML o de texto. Al final de la ejecución, los cambios deben reflejarse en el archivo. **Bonificación de 1 punto si usan XML.**

3. Notas Importantes

- Solamente se pueden usar implementaciones de pilas con arreglos o listas ligadas vistas en clase o ayudantías.
- No hay restricciones sobre el orden cómo se inserten los caracteres que forman las distintas cadenas en la pila, la única condición es que al hacer `popString()` se recupere la cadena original sin ninguna alteración. Por lo que la representación de la pila también se podría ver como:

10
'n'
'o'
'i'
'c'
'i'
's'
'o'
'p'
'e'
'R'

4. Reglas Importantes

- Cumple con los lineamientos de entrega.
- Tu programa debe ser robusto.
- Todos los archivos deberán contener nombre y número de cuenta.
- Tu código debe estar comentado. Esto abarca clases, interfaces, atributos, métodos, etc.
- Para cada clase e interfaz solicitada, crea un nuevo archivo.
- Utiliza correctamente las convenciones para nombrar variables, constantes, clases y métodos.