



INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY

Programación de estructuras de datos y
algoritmos fundamentales

Actividad individual: Reflexión de Actividad
Integral estructura de datos lineales

Presentado por:
Misael Jiménez Hernández A01706158

Primera Sección del Código

Para la implementación de la primera sección del código utilice pilas. La aplicación de pilas para la generación de este código me permitió optimizar la velocidad y el espacio de memoria. Así mismo, debido a que es un programa que solo accede al último valor, la mejor forma para ser implementado es con pilas.

El código consiste de la siguiente forma: El programa mete todos los símbolos de apertura ([, (, {) en una pila y en caso de existir una cerradura (],), }) se revisará si el elemento que se encuentra hasta arriba de la pila hace par con dicho símbolo. En caso de no existir par con el símbolo, regresa false. Este proceso funciona de la misma manera para cada símbolo de apertura ([, (, {)

Complejidad $O(n)$

La complejidad de la primera Sección del Código es $O(n)$. La complejidad $O(n)$ representa la complejidad de una función que aumenta linealmente y en proporción directa al número de entradas.

Segunda Sección del Código

Para la implementación de esta segunda sección del Código, utilice pilas y filas. En esta sección fueron seguidas las instrucciones proporcionadas en GitHub por el profesor. El código consiste de la siguiente forma: Si se encontraba un número, sería insertado en la fila, después, si encontraba un paréntesis izquierdo, sería insertado en la pila. Por otro lado, si se encontraba un operador (+, -, *, /), y mientras que la pila no esté vacía y además el tope de la pila sea diferente al paréntesis izquierdo se realiza lo siguiente: Si el elemento del tope de la pila tiene mayor precedencia que cualquiera de los operadores entonces, se saca el elemento que se encuentra en la pila y es insertado en la fila resultante. En caso contrario se termina el ciclo. Si llegase a existir un operador de mayor precedencia, la instrucción es la misma.

En caso de encontrarse un paréntesis derecho, mientras que la pila no esté vacía y el tope de la pila sea diferente al paréntesis izquierdo, se saca el elemento del tope de la pila y es insertado en la fila. Si la pila no está vacía, se remueve el paréntesis izquierdo del tope de la pila. Si la pila no está vacía, se remueven uno por uno todos los elementos de la pila y son insertados en la fila resultante. Finalmente se devuelve una cadena conformada de todos los elementos de la fila resultante.

Complejidad $O(n^2)$

Complejidad cuadrática. Aparece en bucles o ciclos doblemente anidados. Si n se duplica, el tiempo de ejecución aumenta cuatro veces.

Conclusiones Generales

En conclusión, esta actividad realizada me permitió integrar todos los conocimientos anteriormente vistos en clase con respecto al uso de pilas y filas. He de mencionar, que ha sido la primera vez que tengo contacto con la materia de Estructura de Datos así como del tema realizado en esta ocasión, por lo que a medida que amplió mi horizonte de conocimientos mejoro mi capacidad de organizar datos en una computadora para que puedan ser utilizados de manera eficiente. Aprendí a encontrar ventajas en el uso de filas y pilas en lugar de listas o vectores que no siempre suelen ser los óptimos para cada situación.