

Práctica 1 de Lenguajes – Paradigma Imperativo

Ejercicios

NOTAS:

- Los siguientes ejercicios pretenden que el programador se enfoque en el algoritmo que resuelve el problema y no en la interacción con el usuario para obtener datos, así que se pueden trabajar con datos quemados y haciendo pruebas de funciones/métodos controladas por el programador en el main de los programas si se requiere.
 - Los ejercicios marcados con flechas azules son los obligatorios para subir al repositorio durante la semana de trabajo extraclase
- 1) Haga un programa que cuente e indique el número de caracteres, el número de palabras y el número de líneas de un texto cualquiera (obtenido de cualquier forma que considere). Considere hacer el cálculo de las tres variables en el mismo proceso.
 - 2) Escriba el programa más eficiente que pueda para imprimir en pantalla la siguiente figura:

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

Para dicho fin, escriba y use una función que reciba de parámetro la cantidad de elementos de la línea del centro, la cual debe ser impar positiva.

- 3) Escriba una función que permita rotar una secuencia de elementos N posiciones a la izquierda o a la derecha según sea el caso en función al parámetro que se reciba. Los parámetros deben ser un puntero a un arreglo previamente creado, la cantidad de movimiento de rotación y la dirección (0 = izq y 1 = der).

A partir de esta función, escriba un programa que haga varias rotaciones cualesquiera de una secuencia de elementos previamente creada que sea inmutable. Al final debe imprimirse el resultado de cada rotación además de la secuencia original.

Un ejemplo de rotación puede ser el siguiente:

Secuencia Original = [a, b, c, d, e, f, g, h,]

Cantidad de rotaciones = 3

Dirección = izq

Secuencia final rotada = [d, e, f, g, h, a, b, c] ← *Nótese que cada iteración, el elemento más a la izquierda pasó a formar parte del final de la secuencia, si la rotación fuera a la derecha, sería lo contrario*

- 4) Escriba un programa que utilice una estructura y un arreglo/slice para almacenar en memoria un inventario de una tienda que vende zapatos. Cada zapato debe contar con la información de su modelo (marca), su precio y la talla del mismo que debe ir únicamente del 34 al 44. La estructura debe llamarse "calzado". El programa debe poder almacenar la información "quemada" para 10+ zapatos del inventario y un stock de N cantidad de unidades de dicho zapato (quiere decir por ejemplo que puede existir en inventario el zapato marca Nike, talla 42 y que cuesta 50000 colones, pero además que puede existir no solo un par de esos, si no N pares del mismo zapato).

Haga una función que venda zapatos (eliminando del stock cada vez que haya venta e indicando que no se puede vender porque ya o hay stock) y pruébela dentro del main haciendo las ventas que usted considere para demostrar su funcionamiento.

- 5) Implemente un programa que, utilizando pilas creadas a base de slices, realice un evaluador de expresiones matemáticas expresadas en notación infija. Se debe probar el programa a partir de al menos 3 expresiones con diferentes características dadas por el programador (el main tendrá tres llamadas al evaluador con 3 expresiones diferentes)

Como ejemplos de expresiones y de evaluación se adjuntan las siguientes figuras. De ser necesario se puede ahondar más en las expresiones en infijo y en el algoritmo de evaluación utilizando pilas.



