Capítulo 4.- Pilas

Cuestionario

4.1 Una pila es una estructura lineal
a) Verdadero
b) Falso

4.2 Una pila solamente tiene una puerta de acceso de datos
a) Verdadero
b) Falso

4.3 Una pila es estática
a) Verdadero
b) Falso

4.4 Para usar una pila, debe definirse su tamaño máximo
a) Verdadero
b) Falso

4.5 El tamaño de una Pila puede crecer conforme se insertan nuevos elementos
a) Verdadero
b) Falso

4.6 El primer índice de una Pila en C# siempre es 1
a) Verdadero
b) Falso

4.7 Una pila no necesita controlar la cantidad de elementos que almacena
a) Verdadero
b) Falso

4.8 El comportamiento de una pila es *LIFO* (*Last Input - First Output*)
a) Verdadero
b) Falso

4.9 El comportamiento de una pila es *FILO* (*First Input - Last Output*)
a) Verdadero
b) Falso

4.10 El método `Peek()` elimina el primer nodo de una pila genérica de tipo `Stack`
a) Verdadero
b) Falso

4.11 ¿Cuál es la operación de la Pila que inserta un nuevo dato?
a) Push()
b) Pop()
c) EstaLlena()
d) EstaVacia()

4.12 ¿Cuál es la operación de la Pila que elimina un dato?
a) Push()
b) Pop()
c) EstaLlena()

d) EstaVacia()

4.13 ¿Cómo se detecta si una Pila está llena?
Considere que el primer índice del arreglo es cero
a) Si Top = Max - 1
b) Si Top = Max
c) Si Top > Max
d) Si Top + 1 = Max

4.14 ¿Cómo se detecta si una Pila esta vacía?
Considere que el primer índice del arreglo es cero
a) Si Top = Max - 1
b) Si Top = Max
c) Si Top = 0
d) Si Top = -1

4.15 ¿Qué pasa si se trata de asignar a una pila más valores de los que le caben?
a) Nada
b) Posible funcionamiento incorrecto
c) Ocurre un error
d) Otros datos pueden sobre-escribirse

4.16 ¿Cuál es la clase genérica para administrar una pila?
a) ArrayList
b) Stack
c) Queue
d) List

4.17 ¿Cómo se vacía una pila?
a) Haciendo Top = Max
b) Haciendo Top = -1
c) Haciendo Top = 0
d) Haciendo Max = 0

4.18 Método de la clase genérica `Stack` que devuelve la cantidad de datos almacenados en la pila
a) Push
b) Pop
c) Peek
d) Count

4.19 Método de la clase genérica `Stack` que elimina el dato ubicado en la puerta de la pila
a) Delete
b) Pop
c) Peek
d) Remove

4.20 Método de la clase genérica `Stack` para insertar un dato en la pila
a) Add
b) Push
c) Insert

d) Agregate

Ejercicios

4.1. Modifique el método `Push()` de la aplicación de consola para que no permita duplicados.

4.2. Suponga que tiene una pila de tamaño 7 que almacena cadenas. Dibuje la pila una vez realizada cada una de las siguientes operaciones (incluyendo el `Top` y `Max`).

a) Push("Rodolfo")
b) Push("Manuel")
c) Push("Emiliano")
d) Pop()
e) Push("Alex")
f) Pop()

4.3. Suponga que tiene una pila que almacena datos numéricos enteros positivos y negativos. Diseñe un método que elimine los números negativos de la pila.

4.4. Copie todos los datos de la pila A en la pila B. Utilice los métodos declarados en la `ClasePila` de la aplicación de consola.

4.5. Modifique el método `Push()` de la aplicación de consola para que ordene los datos al momento de insertarlos.

4.6. Agregue un método a la `ClasePila` de aplicación de consola para que elimine los datos duplicados de la pila.

4.7. Agregue un método a la `ClasePila` de la aplicación de consola para que reciba como parámetro un dato y lo elimine de la pila.