**Ejercicio #1**

Sean **nodo1**, **nodo2** y **nodo3** tres nodos consecutivos de una lista (nodo2 es el siguiente a nodo1 y nodo3 es el siguiente a nodo2).

Analice el siguiente fragmento de código (utilice dibujos o diagramas para clarificar qué es lo que sucede):

Nuevo nodo otroNodo

otroNodo.siguiente nodo1

nodo2.siguiente nodo3

a) Inserta “otroNodo” en la lista, quedando como anterior a nodo1.

b) Inserta “otroNodo” en la lista, quedando entre nodo1 y nodo2.

c) Elimina nodo2 de la lista.

d) No tiene ningún efecto sobre la lista.

**Ejercicio #2**

Sean **nodo1**, **nodo2** y **nodo3** tres nodos consecutivos de una lista (nodo2 es el siguiente a nodo1 y nodo3 es el siguiente a nodo2).

Analice el siguiente fragmento de código (utilice dibujos o diagramas para clarificar qué es lo que sucede):

Nuevo nodo otroNodo

otroNodo nodo1.siguiente

nodo1.siguiente nodo3

a) Inserta “otroNodo” en la lista, quedando como anterior a nodo1.

b) Inserta “otroNodo” en la lista, quedando entre nodo1 y nodo2.

c) Elimina nodo2 de la lista.

d) No tiene ningún efecto sobre la lista.

**Ejercicio #3**

Sean **nodo1**, **nodo2** y **nodo3** tres nodos consecutivos de una lista (nodo2 es el siguiente a nodo1 y nodo3 es el siguiente a nodo2).

Analice el siguiente fragmento de código (utilice dibujos o diagramas para clarificar qué es lo que sucede) y responda las preguntas proyectadas en pantalla:

Nuevo nodo otroNodo

otroNodo.siguiente nodo1.siguiente

nodo1.siguiente otroNodo

a) Inserta “otroNodo” en la lista, quedando como anterior a nodo1.

b) Inserta “otroNodo” en la lista, quedando entre nodo1 y nodo2.

c) Elimina nodo2 de la lista.

d) Dará error en tiempo de ejecución si nodo1 es el primero o nodo3 es el último.

**Ejercicio #4**

Analice el siguiente fragmento de código (utilice dibujos o diagramas para clarificar qué es lo que sucede) y responda las preguntas proyectadas en pantalla:

Nuevo nodo otroNodo

Nuevo nodo nodoActual

nodoActual primero

mientras nodoActual <> nulo hacer

nodoActual nodoActual.siguiente

fin mientras

nodoActual.siguiente otroNodo

a) Inserta correctamente “otroNodo” en la lista, quedando como último nodo.

b) Inserta correctamente “otroNodo” en la lista, quedando como primer nodo.

c) El algoritmo está mal hecho, ya que dará error en tiempo de ejecución si la lista está vacía.

d) El algoritmo está mal hecho, ya que dará siempre error en tiempo de ejecución.

**Ejercicio #5**

Analice el siguiente fragmento de código (utilice dibujos o diagramas para clarificar qué es lo que sucede) y responda las preguntas proyectadas en pantalla:

Nuevo nodo otroNodo

Nuevo nodo nodoActual

nodoActual primero

mientras nodoActual.siguiente <> nulo hacer

nodoActual nodoActual.siguiente

fin mientras

nodoActual.siguiente otroNodo

a) Inserta correctamente “otroNodo” en la lista, quedando como último nodo.

b) Inserta correctamente “otroNodo” en la lista, quedando como primer nodo.

c) El algoritmo está mal hecho, ya que dará error en tiempo de ejecución si la lista está vacía.

d) El algoritmo está mal hecho, ya que dará siempre error en tiempo de ejecución.