|  |
| --- |
| **Ingeniero en Computación**  **2018-2**  **Materia:** Algoritmos y Estructuras de Datos / Clave 12098  **Alumno:** Carlos Daniel Orozco Bravo  **Matrícula:** 351285  **Maestro:** Pedro Núñez Yépiz  **Práctica No.:** 4  **Tema – Unidad:** Colas  **Ensenada Baja California a 8 de diciembre del 2018** |

|  |
| --- |
| 1. **INTRODUCCIÓN** |
| En esta practica desarrollaremos un programa de computadora en lenguaje c, capaz de utilizar estructura de datos, para poder manipular una cola. |

|  |
| --- |
| 1. **COMPETENCIA** |
| En esta práctica se busca que el alumno desarrolle la competencia, de poder hacer y manipular colas, desarrollado en un lenguaje c, combinando estructura de datos. |

|  |
| --- |
| 1. **FUNDAMENTOS** |
| **Definición de cola**  Una cola es un tipo especial de lista abierta en la que sólo se pueden insertar nodos en uno de los extremos de la lista y sólo se pueden eliminar nodos en el otro. Además, como sucede con las pilas, las escrituras de datos siempre son inserciones de nodos, y las lecturas siempre eliminan el nodo leído.  Este tipo de lista es conocido como lista FIFO (First In First Out), el primero en entrar es el primero en salir.  El símil cotidiano es una cola para comprar, por ejemplo, las entradas del cine. Los nuevos compradores sólo pueden colocarse al final de la cola, y sólo el primero de la cola puede comprar la entrada.  El nodo típico para construir pilas es el mismo que vimos en los capítulos anteriores para la construcción de listas y pilas:  struct nodo {  int dato;  struct nodo \*siguiente;  };  **Declaraciones de tipos para manejar colas en C**  Los tipos que definiremos normalmente para manejar colas serán casi los mismos que para manejar listas y pilas, tan sólo cambiaremos algunos nombres:  typedef struct \_nodo {  int dato;  struct \_nodo \*siguiente;  } tipoNodo;    typedef tipoNodo \*pNodo;  typedef tipoNodo \*Cola;  tipoNodo es el tipo para declarar nodos, evidentemente.  pNodo es el tipo para declarar punteros a un nodo.  Cola es el tipo para declarar colas.    Es evidente, a la vista del gráfico, que una cola es una lista abierta. Así que sigue siendo muy importante que nuestro programa nunca pierda el valor del puntero al primer elemento, igual que pasa con las listas abiertas. Además, debido al funcionamiento de las colas, también deberemos mantener un puntero para el último elemento de la cola, que será el punto donde insertemos nuevos nodos.  Teniendo en cuenta que las lecturas y escrituras en una cola se hacen siempre en extremos distintos, lo más fácil será insertar nodos por el final, a continuación del nodo que no tiene nodo siguiente, y leerlos desde el principio, hay que recordar que leer un nodo implica eliminarlo de la cola. |

|  |
| --- |
| 1. **PROCEDIMIENTO** |
| Hacer un programa de colas usando estructuras dinámicas. |

|  |
| --- |
| 1. **RESULTADOS Y CONCLUSIONES** |
| **Conclusiones:**  Se termino desarrollando un programa capaz de manipular y simular una cola con estructura de datos. |

|  |
| --- |
| 1. **ANEXOS** |
| **Menú.**    **Menú pilas.**    **Con el 1 se agrega a la pila.**      **Con el 2 se saca de la pila.**        **Menú cola.**    **Con el uno se agrega a la cola.**      **Con el 2 se saca de la pila.** |

|  |
| --- |
| 1. **REFERENCIAS** |
| **JAVA SWING,**  Marc Loy, Robert Eckstein, Dave Wood, James Elliott, Brian Cole.  second edition.  O’Reilly. 2002.  **Como programar en C/C++**  H.M. Deitel/ P.J. Deitel  Segunda edición  Editorial: Prentice Hall.  ISBN:9688804711  **ESTRUCTURAS DE DATOS CON C Y C++**  YEDIYAH LANGSAM; MOSHE AUGENSTEIN  Segunda edición  Editorial: Prentice Hall.  ISBN: 9789688807989  **Ligas de Referencia:**  http://c.conclase.net/edd/index.php?cap=003#inicio |