|  |
| --- |
| **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.:** 5  **Tema – Unidad:** Listas  **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 listas enlazadas. |

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

|  |
| --- |
| 1. **FUNDAMENTOS** |
| **Definición de listas.**  La forma más simple de estructura dinámica es la lista abierta. En esta forma los nodos se organizan de modo que cada uno apunta al siguiente, y el último no apunta a nada, es decir, el puntero del nodo siguiente vale NULL.  En las listas abiertas existe un nodo especial: el primero. Normalmente diremos que nuestra lista es un puntero a ese primer nodo y llamaremos a ese nodo la cabeza de la lista. Eso es porque mediante ese único puntero podemos acceder a toda la lista.  Cuando el puntero que usamos para acceder a la lista vale NULL, diremos que la lista está vacía.  El nodo típico para construir listas tiene esta forma:  struct nodo {  int dato;  struct nodo \*siguiente;  };  En el ejemplo, cada elemento de la lista sólo contiene un dato de tipo entero, pero en la práctica no hay límite en cuanto a la complejidad de los datos a almacenar.  **Declaraciones de tipos para manejar listas en C**  Normalmente se definen varios tipos que facilitan el manejo de las listas, en C, la declaración de tipos puede tener una forma parecida a esta:  typedef struct \_nodo {  int dato;  struct \_nodo \*siguiente;  } tipoNodo;    typedef tipoNodo \*pNodo;  typedef tipoNodo \*Lista;  tipoNodo es el tipo para declarar nodos, evidentemente.  pNodo es el tipo para declarar punteros a un nodo.  Lista es el tipo para declarar listas, como puede verse, un puntero a un nodo y una lista son la misma cosa. En realidad, cualquier puntero a un nodo es una lista, cuyo primer elemento es el nodo apuntado.    Es muy importante que nuestro programa nunca pierda el valor del puntero al primer elemento, ya que si no existe ninguna copia de ese valor, y se pierde, será imposible acceder al nodo y no podremos liberar el espacio de memoria que ocupa. |

|  |
| --- |
| 1. **PROCEDIMIENTO** |
| Hacer un programa de listas abiertas o simples, con estructuras de datos dinamicas,  usando una estructura básica. |

|  |
| --- |
| 1. **RESULTADOS Y CONCLUSIONES** |
| **Conclusiones:**  Se termino desarrollando un programa capaz de manipular listas abiertas con estructura básica. |

|  |
| --- |
| 1. **ANEXOS** |
| **Menú.**    **Menú lista estructura simple.**    **Con el 1 se agrega. (se agregaron 5).**    **Con el 4 se imprime.**    **Con el 3 se busca. (se busca el 28).**    **Se elimina con el 2. (se elimina el 28)**      **Menú de lista estructura alumnos.**    **Con el 1 se agrega. (se agregaron 5).**    **Con el 4 se imprime.**    **Se busca con el 3. (se busca el 300675)**    **Con el 2 se elimina.** |

|  |
| --- |
| 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=001#inicio |