



Algoritmos y Estructuras de Datos II

Trabajo Práctico 2 - Parte 1

Listas enlazadas. Implementadas con punteros.

OBJETIVOS

- Consolidar los conceptos de *punteros*, y *memoria dinámica*.
- Realizar prácticas sobre contenidos de las variables utilizando punteros.
- Aprender a implementar listas con punteros.
- Fortalecer la práctica de las operaciones que se pueden realizar con *listas enlazadas*.

COMPETENCIAS

- Identificar, formular y resolver problemas mediante programación.
- Utilizar de manera efectiva técnicas y herramientas de aplicación para desarrollar software.
- Desempeñarse de manera efectiva en equipos de trabajo.
- Aprender en forma continua, autónoma y de manera colaborativa.

METODOLOGÍA

- El alumno deberá resolver individualmente los ejercicios propuestos.
- El alumno deberá codificar las soluciones en el lenguaje de programación C.
- Realizar consultas a través del canal de slack correspondiente a su comisión ó del aula virtual de la asignatura.

DURACIÓN

Según planificación de la asignatura se deberán utilizar para la resolución de los ejercicios de esta parte, 2 (dos) clases prácticas.

Nota

Tener en cuenta que se deben realizar las operaciones básicas necesarias para la manipulación de la lista con punteros en todos los ejercicios.

EJERCICIOS DE LISTAS ENLAZADAS

1. Crear un tipo de dato **tListaProductos** para un nodo que permita almacenar los siguientes datos de un producto: *codProducto*, *descripción*, *precio unitario*. Luego, declarar un puntero a una variable de ese tipo, y crear el primer nodo de la lista a partir del ingreso de datos por teclado. Por último, mostrar los datos del primer nodo almacenado.
2. Modificar el ejercicio 1 para que se puedan realizar todas las operaciones básicas para el manejo de una lista enlazada:
 - a) Generar o inicializar la lista.
 - b) Función para determinar si una lista está vacía.
 - c) Insertar el primer producto (nodo) de la lista.
 - d) Insertar un producto al principio de la lista.
 - e) Eliminar el primer producto de la lista.
 - f) Insertar un producto en una determinada posición dentro de la lista.
 - g) Eliminar un producto de una determinada posición de la lista.
 - h) Visualizar todos los datos de los productos que están en la lista.
3. El departamento de alumnado necesita trabajar con los datos de los alumnos de la materia AEDII, para ello implementa una lista simplemente enlazada, donde cada nodo guarda el número de libreta universitaria y el nombre del alumno. Se pide crear las funciones necesarias para:
 - a) Generar la lista.
 - b) Insertar el primer alumno (nodo) de la lista.
 - c) Insertar alumnos al principio de la lista.
 - d) Eliminar el primer alumno de la lista.
 - e) Insertar un alumno en una determinada posición dentro de la lista.
 - f) Eliminar un alumno de una determinada posición de la lista.
 - g) Visualizar todos los datos de los alumnos que están en la lista.

Además, crear una función que retorne la cantidad de alumnos que contiene la lista e implementar un menú principal que contenga todos los ítems anteriormente solicitados.
4. Un Instituto médico realiza una encuesta sobre nutrición. Para procesar los datos de los encuestados requiere un sistema, que implemente una lista enlazada. Para ello se capturan los siguientes datos: código de sexo (F. Femenino, M. Masculino), código de rango de edad (01. Menor de 15 años, 02. Entre 15 y 30 años, 03. Más de 30 años), peso y altura. El sistema debe calcular el IMC (índice de masa corporal) antes de insertar un elemento en la lista, y almacenar la clasificación del IMC (string) como **parte contenida de cada nodo**.

Fórmula del IMC = kg / m^2

Clasificaciones	Mujeres	Varones
Peso bajo	< 20	< 19
Peso normal	20 - 25	19 - 24
Sobrepeso	>25	>24

Se solicita realizar las funciones necesarias para que los usuarios del sistema puedan obtener los siguientes informes:

- a) La cantidad total de encuestados.
- b) El promedio de pesos de las personas de peso bajo, que sean menores de 15 años.
- c) Un listado de los datos correspondientes a las personas que tienen sobrepeso.