Actividad: Estructura de Datos 02/09/24
Equipo No.1

Realiza los siguientes pseudocódigos para insertar elementos:

1) Insertar un dato en una lista en orden ascendente:

```
Funcion ascendente (nodo, lista)
  x=0
  Para i=0 hasta i<lista.[ize * con incremento en 1 hacer
    Si (nodo.data<lista[i].data)
      insert_prev(nodo, lista[i], lista)
      x=1
      Fin Para
    Si no si (nodo.data==lista[i].data)
      x=1
      Fin Para
    Fin Si
  Fin Para
  Si no (x)
    insert_next(nodo, lista.tail, lista)

  Fin Si
Fin Funcion
```

2) Insertar un dato en una lista en orden descendente:

```
Funcion descendente(nodo, lista)
  x=0
  Para i=0 hasta i<lista.size y x==0 con i++ hacer
    Si (nodo.data>lista[i].data)
      insert_prev(nodo, lista[i], lista)
      x=1
    Si no si (nodo.data==lista[i].data)
      x=1
    Fin Si
  Fin Para
  Si no (x)
    insert_next(nodo, lista.tail, lista)
  Fin Si
Fin Funcion
```

3) Inserción básica, insertar un nodo al fina de la lista:

```
Llamar insert_next(nodo, lista.tail, lista)

Funcion insert_next(nodo, actual, lista)
  si lista.size==0
    l.tail=nodo //se puede omitir
    l.head=nodo
      nodo.next=NULL

  Si no
     nodo.next=actual.next
     actual.next = nodo
  Fin Si

  Si (nodo.next=NULL)
    l.tail=nodo
  Fin Si
  lista.size=lista.size+1
Fin Funcion
```

4) Dada una lista enlazada (sencilla) l. Crear una función que devuelva una lista (l_1) tal que l_1 contenga los datos impares de l.

```
Funcion impar(lista, l1)
   l1.size=0
   l1.tail=NULL
   ult=NULL
   l1.head=NULL

   Para i=0 hasta i<lista.size con incremento 1 hacer
      Si (lista[i].data%2 !=0)
         insert_next(lista[i], ult, l1)
         ult= l1[sis.size-1]
      Fin Si
   Fin Para

   Retornar l1  //No necesario con Punteros
Fin Funcion
```

5) Dada una lista enlazada (sencilla) l. Crear una función que devuelva una lista (l1) tal que los nodos almacenen un dato mayor al promedio.

```
Funcion listaP(l, l1)
  var=0
  ult=NULL
  l1.head=ult
  l1.tail=NULL
  l1.size=0

  Para i=0 hasta i<l.size incremento en 1 hacer
     var=var+l[i].data
  Fin para

  Var = Ceil(var/l.size)
* nodo.
  Para i=0 hasta i<l.size incremento en 1 hacer
     nodo.data=var
     var=var+1
     insert_next(nodo, ult, l1)
     ult=l1[l1.size-1]
  Fin para
Fin Funcion
```

Nota 1), en los bucles "para", la palabra hasta puede usarse como un mientras.

Nota 2), en el 5), no returna nada porque trabaja con punteros.

Nota 3), en el 1) y 2), utiliza las funciones insert_next e insert_prev de la lista doblemente ligada trabajada de tarea.

Integrantes:

- Hernández Peña Angel Adrian
- Siles Bazaldua Humberto Tomas
- Ruiz Garcia Emiliano
- Franco Mitzi Natalie Gabrielle
- Dominguez Lira Estefani Michalle
- Aguilar Buendia Bruno