Ambiente

asd = registro

// datos

Dato: alfanumerico

// punteros

prox: puntero a asd

finsi

p, a: puntero a asd

Proceso

Prim:= nil

ESCRIBIR("ingrese un valor para guardar en lista, para finalizar ingrese \*")

LEER (valor)

MIENTRAS (valor <> '\*') HACER

NUEVO (p)

\*p.Dato:= valor

\*p.Prox:= nil

SI Prim = nil ENTONCES

Prim = p

SINO

\*a.Prox:= p

FIN SI

a:=p

ESCRIBIR("ingrese un valor para guardar en lista, para finalizar ingrese \*")

LEER(valor)

FIN MIENTRAS

Accion Ejercicio\_4\_4(Prim: puntero a nodo) es

Ambiente

nodo = registro

dato: entero

prox: puntero a nodo

finregistro

salida, m, p: puntero a nodo

Proceso

m:= Prim

salida:= nil

MIENTRAS (m <> nil ) HACER

NUEVO(p)

\*p.Dato:= \*m.dato

\*p.Prox:= salida

salida:= p

m:= \*m.prox

FIN MIENTRAS

FinAccion

// Casos de eliminacion

Condicion Acciones

UNICO ELEMENTO p = prim prim:= nil (o prim:= \*p.prox )

a = nil

PRIMER ELEMENTO misma prim:= \*p.prox

ELEMENTO INTERMEDIO p<> nil \*a.prox:= \*.prox

a<> nil

p.dato = dato

ULTIMO ELEMENTO p.prox = nil \*a.prox:= nil

a <> nil

p.dato = dato

SI (\*q.dato = valor) ENTONCES

SI (a = nil) ENTONCES

Prim = \*q.prox

SINO

\*a.prox:= \*q.prox

FIN SI

DISPONER(q)

SINO

Escribir ('Error, el valor no existe')

FIN SI

TAREA DE LA CLASE:  
Se tiene una lista con los nombres de los clientes de una compañía de telefonía celular, los cuales pueden aparecer repetidos en la lista, si tienen registrado más de un número telefónico.

La compañía para su próximo aniversario desea enviar un regalo a sus clientes, sin darle mas de un regalo a cada cliente. En una lista circular se almacenan los regalos disponibles. Se desea generar un algoritmo que cree una nueva lista encadenada ordenada con los nombres y los premios para cada uno (sin repetir los nombres).

RESOLUCION

Se tiene una lista con los nombres de los clientes de una compañía de telefonía celular, los cuales pueden aparecer repetidos en la lista, si tienen registrado más de un número telefónico. La compañía para su próximo aniversario desea enviar un regalo a sus clientes, sin darle mas de un regalo a cada cliente. En una lista circular se almacenan los regalos disponibles. Se desea generar un algoritmo que cree una nueva lista encadenada ordenada con los nombres y los premios para cada uno (sin repetir los nombres).

Accion Personal (primnom: puntero a cliente, primreg: puntero a regalos)

Ambiente

clientes = registro

apynom = AN(30)

teléfono = N(13)

prox = puntero a clientes

fin registro

regalos = registro

regalo = AN(30)

cantidad = entero

prox = puntero a regalos

fin registro

nodo = registro

apynom = AN(30)

regalo = AN(30)

prox = puntero a nodo

fin registro

a, prim, p : puntero a nodo

cli : puntero a clientes

reg : puntero a regalos

existe : booleano

Proceso

prim:= nil

cli:= primcli; reg:= primreg

Mientras (cli <> nil) hacer

existe:= false

q:= prim

Mientras (q <> nil) y (no existe) hacer

Si \*q.apynom = \*cli.apynom entonces

existe:= true

Fin Si

q:=\*q.prox

Fin Mientras

Si no existe entonces

Nuevo (p)

\*p.apynom:= \*cli.apynom

\*p.regalo:= \*reg.regalo

reg:= \*reg.prox

q:= prim; a:= nil

Mientras (q <> nil) y (\*q.apynom < \*p.apynom) hacer

a:= q; q:= \*q.prox

Fin Mientras

Si (a=nil) entonces

\*p.prox:= prim; prim:= p

contrario

\*p.prox:= q; \*a.prox:= p

Fin Si

Fin Si

cli:= \*cli.prox

Fin Mientras

Fin Accion