

Pruebas de escritorio

Clase Lista-circular-Doble

Funcion insertar por cola

Nuevo	Cabeza	ultimo	(!Cabeza)
Datos	cabeza	null	✓
Datos	Apunta a Datos	null	
Datos Ingreso	Apunta a Cabeza	Apunta a Datos 1	
Datos Ingreso	Apunta a Datos	Apunta a Datos	
Datos	Apunta a Datos Ingreso	Apunta a Datos	
Datos Ingreso	Apunta a Datos Ingreso	Apunta a Datos	

Busca, Por Placa Funcion Placa a encontrar en nodo 3

aux	Cabeza	aux.placa==placa	While (aux != cabeza)
Cabeza	Cabeza	X	✓
Nodo 2		X	✓
Nodo 3		✓	✓
Cabeza			X

Placa encontrada

Funcion ActualizarEspacio

4 Nodos

id Espacio = 3

Aux	Cabeza	if (aux->idEspacio == idEspacio)	while (aux != cabeza)
cabeza		1 \neq 3	✓
nodo 2		2 \neq 3	✓
nodo 3		3 = 3	

aux \rightarrow setNombre(nombre)
 aux \rightarrow setApellido(apellido)
 aux \rightarrow setCedula(cedula)
 aux \rightarrow setPlaca(placa)
 aux \rightarrow setMarca(marca)
 aux \rightarrow setColor(color)
 aux \rightarrow setFecha(fecha)

Clase Parqueadero

Funcion ocuparEspacio (int idEspacio)

Variables	Operacion	Resultado
idEspacio = 3	3 / 4 = 0	Fila = 0
columnas = 4	3 % 4 = 3	Columna = 3
11		
Fila = 0 columna = 3 Filas = 2 columnas = 4	verifica posicion valida	True
		En [0][3] marca X

Funcion liberaEspacio (int idEspacio)

Variables	Estado	Accion	Resultado
idEspacio = 3 columnas = 4	3	$3/4 = 0$	Fila = 0
11	3	$3\%4 = 3$	Columna = 3
Fila = 0 columna = 3 filas = 2 columnas = 4	True porque $0 < 280324$	Verifica posicion Valida	True
En [0][3] se cambia a "-"			

Clase Utils

Funcion modifyChar InString

str = abcde target = c

str	target	offset	c	operacion
abcde	c	2	a	$a! = c$ x
abcde	c	2	b	$b! = c$ x
abcde	c	2	c	$c == c$ ✓ $c+2 = e$
abcde	c	2	d	$d! = c$ x
abcde	c	2	e	$e! = c$ x

Clase Lista_Circular_Doble

Funcion cifraString

texto	desplazamiento	resultado	c	Accion
hola	2		h	jola
	2		o	jpla
	2		l	jplk
	2		a	jpkc
	2	jpkc	11	