

Simulación de una empresa de Transporte de automóviles en C++



Álvaro Camacho Martín 48083824R
Óscar Montesano Martínez 53852484P

TAD's implementados en el proyecto

- Pila:

Especificación PILA[VEHICULO]

Operaciones:

{Generadoras}

Pila():pila ; genera una pila vacía de longitud 0

Apilar(Vehiculo): ; Añade un vehículo a la cima de la pila

{Modificadoras}

Parcial Desapilar(): ; Saca el ultimo vehículo añadido a la pila

{Observadoras}

Parcial Mostrar(): Vehiculo; Muestra el último vehículo añadido a la pila

Es_vacia(): Bool; Devuelve True si la pila está vacía

verPila(): ; Muestra la pila, vehículo a vehículo empezando por la cima

get_profundidad(): entero; Devuelve la profundidad de la pila

- Cola:

Especificación COLA[VEHICULO]

Operaciones:

{Generadoras}

Cola():cola ; genera una cola vacía (con ambos nodos a NULL y con longitud 0)

Encolar(Vehiculo): ; Añade un vehículo al final de la cola

{Modificadoras}

Parcial Desencolar(): Vehículo; Saca el ultimo vehículo de la cola (y lo devuelve)

{Observadoras}

Parcial Inicio(): Vehículo; Muestra el primer vehículo de la cola

Get_longitud(): Entero; Devuelve la longitud de la cola, 0 si es vacía

Es_vacia(): Bool; Devuelve True si la cola es vacía

MostrarCola(): Vehiculos; Va mostrando el primer vehículo de la cola uno detrás de otro.

b.2) Para implementar ambos Tipos Abstractos de Datos hemos usado memoria dinámica. Cada TAD consta de dos clases, la suya y el nodo. Un nodo es una celda que contiene un elemento de la estructura de datos además de un puntero que apunta al siguiente nodo. En el caso de Colas, hay además un puntero que apunta al inicio, por donde salen los elementos y otro que apunta al final, por donde entran. La pila solo tiene uno apuntando al nodo que en ese momento sea la cima.

Las dificultades que nos hemos encontrado han principalmente sido trabajar con punteros, ya que nunca lo habíamos hecho, y aunque en java habíamos trabajado con clases en POO, al haber dos ficheros por clase y tantos métodos nos ha costado bastante más corregir los errores.

Hemos conseguido implementar todo lo que pedía el trabajo menos una cosa: mostrar las pilas (camiones) y colas (registros) de las zonas.

b.3) Como hemos mencionado anteriormente, contamos con una clase Nodo por cada TAD. Esta toma como "friend class" a su TAD para que este pueda usarla sin problemas.

Para las colas, tomamos dos punteros a NodoCola, primero y ultimo. Si la cola es vacía apuntan a NULL. Cada vez que queremos insertar un elemento creamos un nuevo NodoCola; Si la cola es vacía tanto primero como ultimo apuntarán a él. Sino, apuntamos el siguiente de ultimo al nuevo nodo y este lo convertimos en ultimo. Finalmente incrementamos la longitud de la cola en uno.

Para las pilas, al crearla la cima apunta a null y la profundidad es 0. A medida que vamos apilando o desapilando vehículos la profundidad varía. Al insertar un vehículo este nodo se convierte en cima y apunta al de debajo.

b.3.1) Cabe destacar que hemos implementado cuatro métodos nuevos en gestor para generar nv, ns, nc y np de manera aleatoria y entre los límites establecidos en el enunciado. Se pueden activar en el método main quitando los comentarios, y de esta forma serán aleatorios para cada simulación. Otro método interesante se trata de crearVehiculo en la clase gestor, el cual recibe como parámetro nv y genera ese número de coches; creamos un objeto coche auxiliar con el constructor, el cual se vale de otros métodos implementados por nosotros los cuales generan sus atributos de manera aleatoria. Los distintos valores que pueden tomar los atributos se encuentran en vectores y se pueden cambiar fácilmente.

b.3.2) El programa comienza con un menú de 8 opciones. Para comenzar la simulación crearemos un numero de coches, ya sea aleatorio (con nuestro método generar nv, o estableciéndolo en el main) o determinado por el usuario, con ambas opciones brindadas por el programa. Una vez haya coches en las colas (Almacenes) podremos observarlos presionando 3. Si queremos llevar coches a alguna cola de zona podremos hacerlo, de igual manera, aleatoriamente o aportando el numero.

Para eliminar los coches de la fabrica basta con pulsar 4.

b.4) Bibliografía: Apuntes de la asignatura.