Paquetes

Com.jared.edd.estructuras

com.jared.edd.estructuras

+destino: Destino +anterior: NodoB +siguiente: NodoB +izquierda: Rama +derecha: Rama

+NodoB(destino:Destino) +getDestino(): Destino +setDestino(destino:Destino): void +getAnterior(): NodoB +setAnterior(anterior:NodoB): void +getSiguiente(): NodoB

+setSiguiente(siguiente:NodoB): void +getIzquierda(): Rama +setIzquierda(izquierda:Rama): void

+getDerecha(): Rama +setDerecha (derecha:Rama): void

GrafoRuta

+primero: NodoGrafo +ultimo: NodoGrafo

+buscar(od:Int): boolean +buscarNombre(id:Int): String

+emptv(): boolean +insertar(destino:Destino): void

+insertarRuta(ruta:Ruta): String +insertarRutaRetorno(ruta:Ruta): void +recorrerGrafo(): String

Rama

primero: NodoB = null cuenta: Int = 0

hoja: boolean = true +Rama (primero:NodoB)

+get.Primero(): NodoB

-setPrimero(primero:NodoB): void -getCuenta(): Int

setCuenta(cuenta:Int): void esHoja(): boolean

setHoja(hoja:boolean): void insertar(nuevo:NodoB): void

ListaRuta

+primero: NodoRuta -ultimo: NodoRuta

recorrer(): String

+buscar(id:Int): boolean buscarCodigo (nombre:Int): Int +buscarCodigo(nombre:String): int buscarD(id:Int): Ruta contiene(id:Int): boolean eliminar(id:Int): boolean +emptv(): boolena +insertar(ruta:Ruta): boolean

ArbolB

+raiz: Rama = null

estaVacio(): boolean Insertar(destino:Destino): void

+dividir(rama:Rama): Object

inserta (nodo:NodoB, rama:Rama): Object obtenerNodo(r:Rama,numero:Int): String obtenerCodigo(r:Rama,numero:Int): Int

TablaHash

-ingresados: Int -size: Int

tabla: Reservacion[]

+inOrden(r:Rama): String

+buscar(nombreUsuario:String): Reservacion

+cambiarSize(): void

esPrimo(numero:Int): boolean

getIngresados(): Int

+getSize(): Int

getTabla(): Reservacion[]

+insertar(reservacion:Reservacion): void

+recorrer(): String

+redireccionar (reservacion: Reservacion, index: Int.

iteracion:Int): void +setIngresados(numero:Int): void

setSize(numero:Int): void +setTabla(reservacion:Reservacion[]): void

MatrizRuta

+primeroFila: NodoFila -+primeroColumna: NodoColumna

+lista: Lista +listaInt: ListaInt

+agregarCabecera(numFila:Int,pais:String): String +agregarRuta(ruta:Ruta): void +buscarPosicion(codigo:Int): Int

+buscarTC(salida:Int,llegada:Int,opcion:Int): Int completar(): String

contarCabecera(): Int

+editar(codigoSalida:Int,codigoLlegada:Int,

costo:Int,tiempo:Int): String editar2(codigoSalida:Int.codigoLlegada:Int. costo:Int,tiempo:Int): String

+eliminar(codigoSalida:Int,codigoLlegada:Int): String

+encontrarCabecera(codigo:Int): boolean

+indexar(): void

+inversorRuta(ruta:Ruta): Ruta

+llenar(): String

+mostrarCabeceras(): String mostrarCabecerasl(): String

+mostrarColumna(): String

+mostrarFila(): String

mostrarLista(): void

retornarColumna(codigo:Int): NodoColumna rutaCorta(codigoSalida:Int,codigoLlegada:Int): Int

+contiene(1:ListaInt): boolean empty(): boolean eget(index:Int): Int insertar(lista:ListaInt): boolean

Lista

insertar(lista:ListaInt): boolean

ListaInt

contiene(l:ListaInt): boolean

get(index:Int): ListaInt

+primero: Nodo

size(): Int

+primero: Nodo

empty(): boolean

ListaDestino

+primero: NodoDestino +ultimo: NodoDestino

+buscar(id:Int): boolean buscarCodigo(nombre:Int): Int +buscarCodigo(nombre:String): int

buscarD(id:Int): Destino contiene(id:Int): boolean

eliminar(id:Int): boolean empty(): boolena

+insertar(destino:Destino): boolean

recorrer(): String

com.jared.edd.nodos

com.jared.edd.nodo

NodoC

+derecha: NodoC +izquierda: NodoC +arriba: NodoC +abajo: NodoC +ruta: Ruta

+NodoC(ruta:Ruta)

NodoColumna

+numColumna: Int
+siguiente: NodoColumna
+abajo: NodoC
+index: Int = 0

+NodoColumna(numColumna:Int)

NodoGrafo

+dato: Destino +siguiente: NodoGrafo +ruta: ListaRuta

+getDato(): Destino
+getRuta(): ListaRuta
+getSiguiente(): NodoGrafo
+setDato(destino:Destino): void
+setRuta(ruta:ListaRuta): void

+setSiguiente(siguiente:NodoGrafo): void

NodoDestino

+dato: Destino +siguiente: NodoDestino

+getDato(): Destino

+getSiguiente(): NodoDestino
+setDato(dato:Destino): void

+setSiguiente(siguiente:NodoDestino): void

NodoRuta

+dato: Ruta

+siguiente: NodoRuta

+getDato(): Ruta

+getSiguiente(): NodoRuta +setDato(dato:Ruta): void

+setSiguiente(siguiente:NodoRuta): void

NodoFila

+numFila: Int
+siguiente: NodoFila
+derecha: NodoC
+index: Int
+pais: String

+NodoFila(numFila:Int,pais:String)

com.jared.edd.recursos

com.jared.edd.recursos

Destinos

+buscar(id:Int): String +buscarN(pais:String): String +graficarArbol(): String +graficarArbolMatriz(): String

+ingresarDestino(id:Int,pais:String): String

+recorrer(): String

Vuelos

Rutas