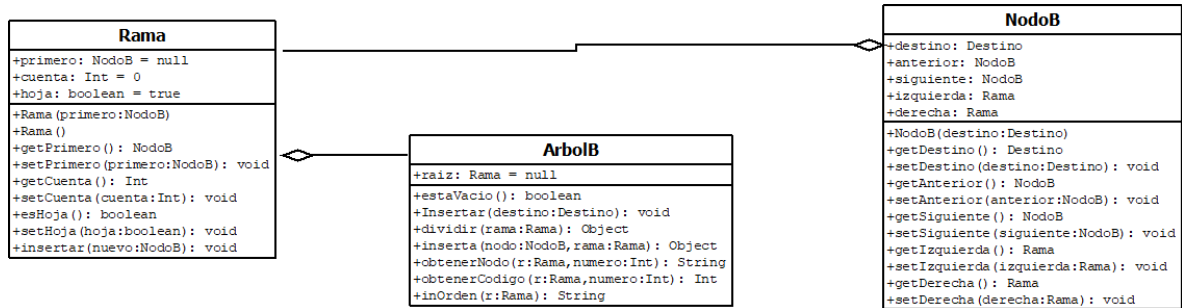


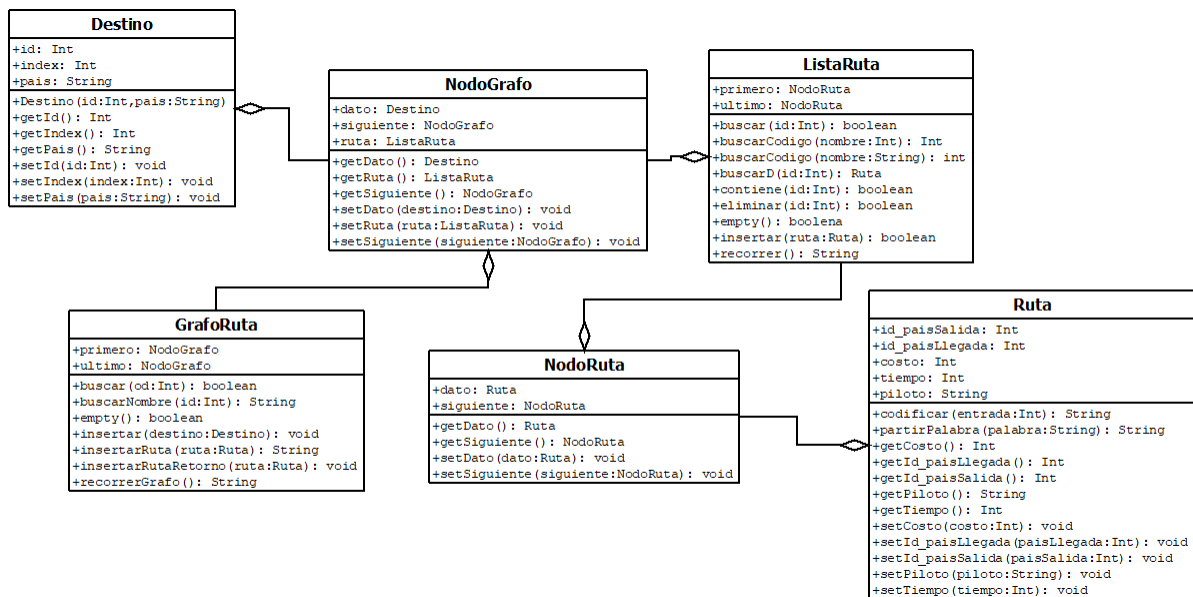
Contenido

Diagrama de clases.....	2
Árbol B.....	2
Grafo.....	2
Destino	2
Matriz	3
Tabla Hash	3
Paquetes.....	4
Com.jared.edd.estructuras	4
com.jared.edd.nodos	5
com.jared.edd.recursos	5

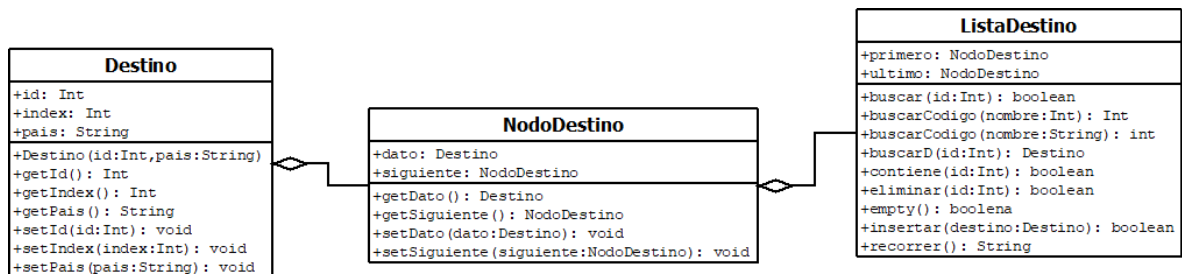
Árbol B



Grafo



Destino



Matriz

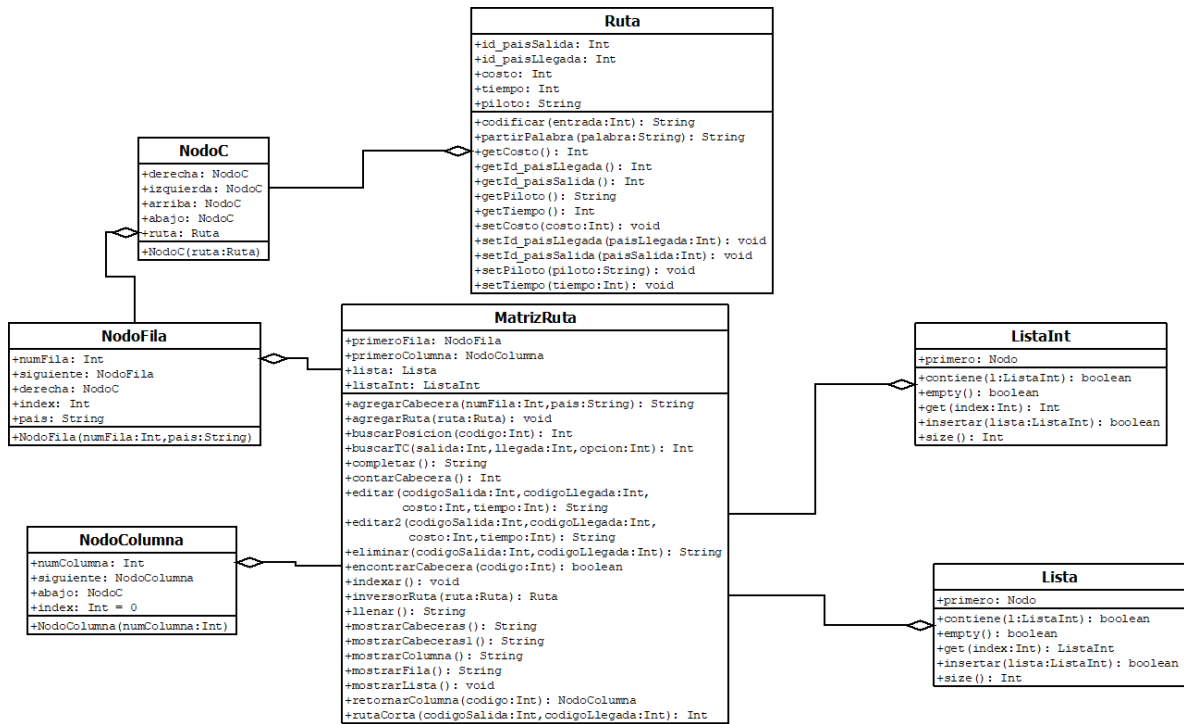
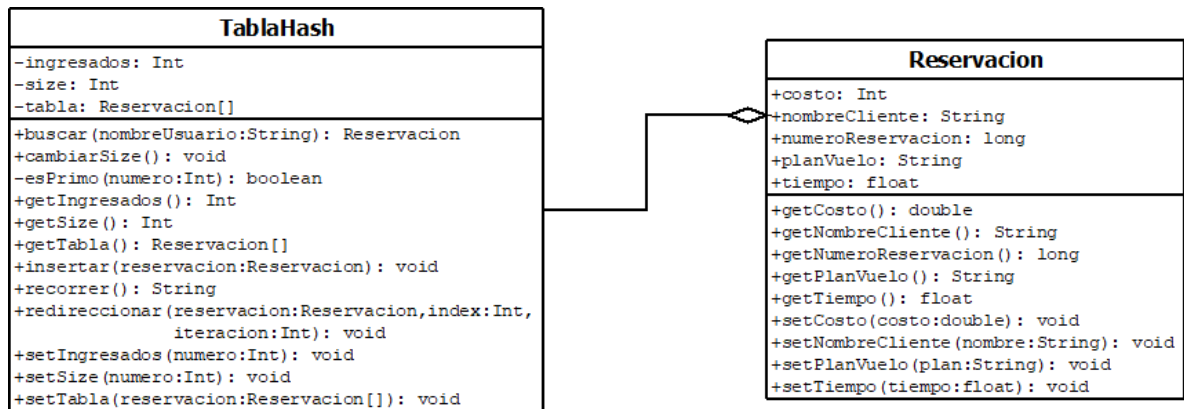


Tabla Hash



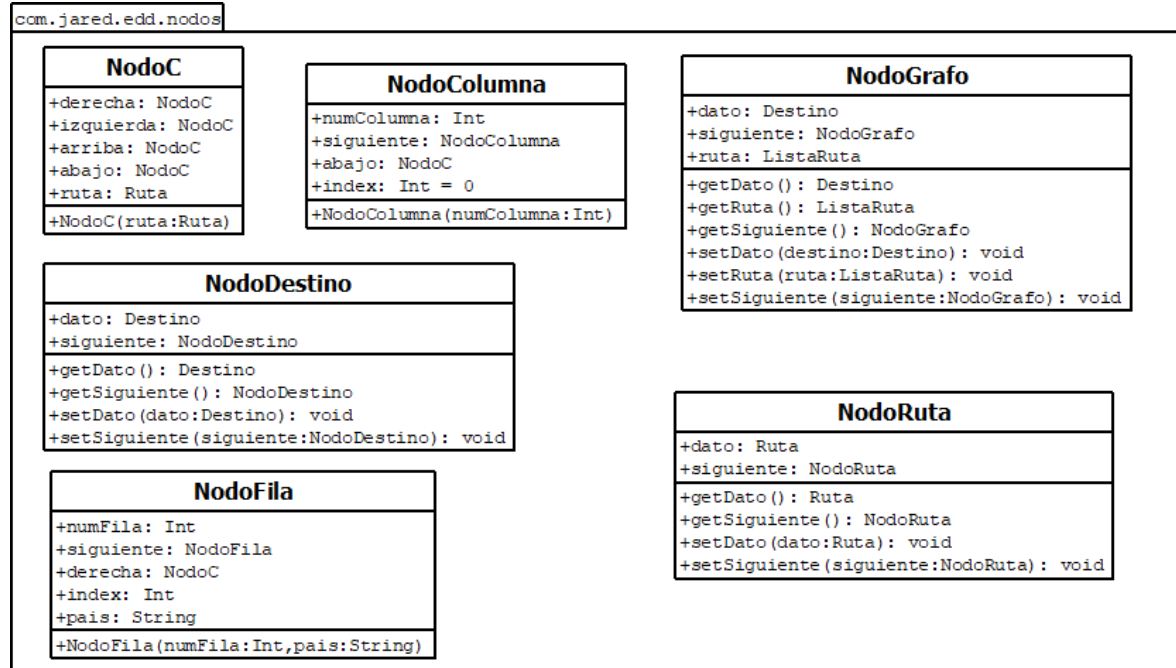
Paquetes

Com.jared.edd.estructuras

com.jared.edd.estructuras

<table><tr><th>NodoB</th></tr><tr><td>+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</td></tr></table>	NodoB	+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	<table><tr><th>Rama</th></tr><tr><td>+primero: NodoB = null +cuenta: Int = 0 +hoja: boolean = true +Rama(primero:NodoB) +Rama() +getPrimero(): NodoB +setPrimero(primero:NodoB): void +getCuenta(): Int +setCuenta(cuenta:Int): void +esHoja(): boolean +setHoja(hoja:boolean): void +insertar(nuevo:NodoB): void</td></tr></table>	Rama	+primero: NodoB = null +cuenta: Int = 0 +hoja: boolean = true +Rama(primero:NodoB) +Rama() +getPrimero(): NodoB +setPrimero(primero:NodoB): void +getCuenta(): Int +setCuenta(cuenta:Int): void +esHoja(): boolean +setHoja(hoja:boolean): void +insertar(nuevo:NodoB): void	<table><tr><th>ArbolB</th></tr><tr><td>+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 +inOrden(r:Rama): String</td></tr></table>	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 +inOrden(r:Rama): String
NodoB								
+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								
Rama								
+primero: NodoB = null +cuenta: Int = 0 +hoja: boolean = true +Rama(primero:NodoB) +Rama() +getPrimero(): NodoB +setPrimero(primero:NodoB): void +getCuenta(): Int +setCuenta(cuenta:Int): void +esHoja(): boolean +setHoja(hoja:boolean): void +insertar(nuevo:NodoB): void								
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 +inOrden(r:Rama): String								
	<table><tr><th>TablaHash</th></tr><tr><td>-ingresados: Int -size: Int -tabla: Reservacion[] +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</td></tr></table>	TablaHash	-ingresados: Int -size: Int -tabla: Reservacion[] +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					
TablaHash								
-ingresados: Int -size: Int -tabla: Reservacion[] +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								
<table><tr><th>GrafoRuta</th></tr><tr><td>+primero: NodoGrafo +ultimo: NodoGrafo +buscar(od:Int): boolean +buscarNombre(id:Int): String +empty(): boolean +insertar(destino:Destino): void +insertarRuta(ruta:Ruta): String +insertarRutaRetorno(ruta:Ruta): void +recorrerGrafo(): String</td></tr></table>	GrafoRuta	+primero: NodoGrafo +ultimo: NodoGrafo +buscar(od:Int): boolean +buscarNombre(id:Int): String +empty(): boolean +insertar(destino:Destino): void +insertarRuta(ruta:Ruta): String +insertarRutaRetorno(ruta:Ruta): void +recorrerGrafo(): String	<table><tr><th>ListaRuta</th></tr><tr><td>+primero: NodoRuta +ultimo: NodoRuta +buscar(id:Int): boolean +buscarCodigo(nombre:Int): Int +buscarCodigo(nombre:String): int +buscarD(id:Int): Ruta +contiene(id:Int): boolean +eliminar(id:Int): boolean +empty(): boolean +insertar(ruta:Ruta): boolean +recorrer(): String</td></tr></table>	ListaRuta	+primero: NodoRuta +ultimo: NodoRuta +buscar(id:Int): boolean +buscarCodigo(nombre:Int): Int +buscarCodigo(nombre:String): int +buscarD(id:Int): Ruta +contiene(id:Int): boolean +eliminar(id:Int): boolean +empty(): boolean +insertar(ruta:Ruta): boolean +recorrer(): String			
GrafoRuta								
+primero: NodoGrafo +ultimo: NodoGrafo +buscar(od:Int): boolean +buscarNombre(id:Int): String +empty(): boolean +insertar(destino:Destino): void +insertarRuta(ruta:Ruta): String +insertarRutaRetorno(ruta:Ruta): void +recorrerGrafo(): String								
ListaRuta								
+primero: NodoRuta +ultimo: NodoRuta +buscar(id:Int): boolean +buscarCodigo(nombre:Int): Int +buscarCodigo(nombre:String): int +buscarD(id:Int): Ruta +contiene(id:Int): boolean +eliminar(id:Int): boolean +empty(): boolean +insertar(ruta:Ruta): boolean +recorrer(): String								
<table><tr><th>MatrizRuta</th></tr><tr><td>+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 +mostrarCabeceras1(): String +mostrarColumna(): String +mostrarFila(): String +mostrarLista(): void +retornarColumna(codigo:Int): NodoColumna +rutaCorta(codigoSalida:Int,codigoLlegada:Int): Int</td></tr></table>	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 +mostrarCabeceras1(): String +mostrarColumna(): String +mostrarFila(): String +mostrarLista(): void +retornarColumna(codigo:Int): NodoColumna +rutaCorta(codigoSalida:Int,codigoLlegada:Int): Int	<table><tr><th>Lista</th></tr><tr><td>+primero: Nodo +contiene(l:ListaInt): boolean +empty(): boolean +get(index:Int): ListaInt +insertar(lista:ListaInt): boolean +size(): Int</td></tr></table>	Lista	+primero: Nodo +contiene(l:ListaInt): boolean +empty(): boolean +get(index:Int): ListaInt +insertar(lista:ListaInt): boolean +size(): Int	<table><tr><th>ListaDestino</th></tr><tr><td>+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(): boolean +insertar(destino:Destino): boolean +recorrer(): String</td></tr></table>	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(): boolean +insertar(destino:Destino): boolean +recorrer(): String
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 +mostrarCabeceras1(): String +mostrarColumna(): String +mostrarFila(): String +mostrarLista(): void +retornarColumna(codigo:Int): NodoColumna +rutaCorta(codigoSalida:Int,codigoLlegada:Int): Int								
Lista								
+primero: Nodo +contiene(l:ListaInt): boolean +empty(): boolean +get(index:Int): ListaInt +insertar(lista:ListaInt): boolean +size(): Int								
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(): boolean +insertar(destino:Destino): boolean +recorrer(): String								
	<table><tr><th>ListaInt</th></tr><tr><td>+primero: Nodo +contiene(l:ListaInt): boolean +empty(): boolean +get(index:Int): Int +insertar(lista:ListaInt): boolean +size(): Int</td></tr></table>	ListaInt	+primero: Nodo +contiene(l:ListaInt): boolean +empty(): boolean +get(index:Int): Int +insertar(lista:ListaInt): boolean +size(): Int					
ListaInt								
+primero: Nodo +contiene(l:ListaInt): boolean +empty(): boolean +get(index:Int): Int +insertar(lista:ListaInt): boolean +size(): Int								

com.jared.edd.nodos



com.jared.edd.recursos

