

Auxiliatura INF-131 “C”

Estructura de Datos y Algoritmos

Univ. Miguel Angel Quispe Mamani

Universidad Mayor de San Andrés

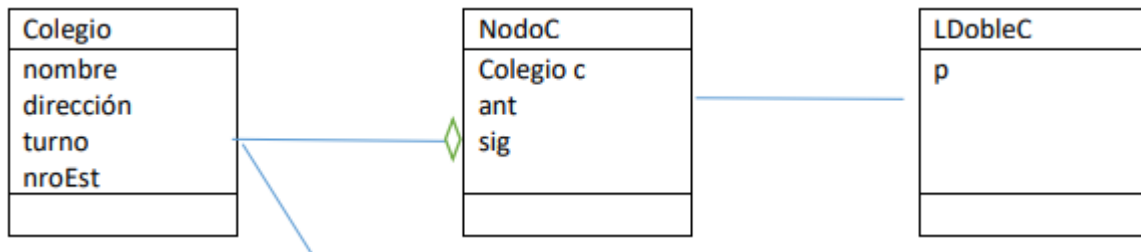
Carrera de Informática

10/04/2023

Nota: Las soluciones estan en el lenguaje **Java**, si tiene complicaciones al pasar el codigo a SeudoCodigo, hágalo saber a su auxiliar, para absolver así sus dudas

Enunciado

Dado el siguiente diagrama de clases:



- Adicionar un nuevo colegio antes del ultimo colegio
- Adicionar k nuevos colegios antes del ultimo colegio
- Adicionar k nuevos colegios después del i esimo

Ejercicio a

```
1 public static void ejercicio1_v2(LDobleC a) {
2     NodoC w = a.getP();
3     if (w == null) return;
4     Colegio u = new Colegio();
5     u.leer();
6     if (w.getSig() == null) a.adiPrincipio(u);
7     else {
8         NodoC aux = a.eliFinal();
9         a.adifinal(u);
10        a.adifinal(aux.getA());
11    }
12 }
```

```
1 //version 2
2 public static void ejercicio1_v1(LDobleC a) {
3     NodoC w = a.getP();
4     if (w == null)
5         return;
6     Colegio u = new Colegio();
7     u.leer();
8     if (w.getSig() == null) {
9         a.adiPrincipio(u);
10    } else {
11        // obtenemos el ultimo nodo
12        while (w.getSig() != null)
13            w = w.getSig();
14        // creamos el nuevo nodo
15        NodoC nue = new NodoC();
16        nue.setA(u);
17        nue.setAnt(w.getAnt());
18        nue.setSig(w);
19        w.getAnt().setSig(nue);
20        w.setAnt(nue);
21    }
22 }
```

Ejercicio b

```
1 public static void adicionarK(LDobleC a, int k) {
2     for (int i = 0; i < k; i++)
3         ejercicio1_v2(a); //puede cualquiera de las 2 versiones del inciso a
4 }
```

```
1 public static void adicionarK2(LDobleC a, int k) {
2     for (int i = 0; i < k; i++)
3         ejercicio1_v1(a); //puede cualquiera de las 2 versiones del inciso a
4 }
```

Ejercicio c

```
1  public static void ejercicio2(LDobleC a, int i, int k) {
2      if (a.nroNodos() < i)
3          return;
4      NodoC w = a.getP();
5      int j = 0;
6      while (w != null) {
7          j = j + 1;
8          if (j == i) {
9              // crear los k nuevos nodos
10             LDobleC b = new LDobleC();
11             b.leer2(k);
12             b.getP().setAnt(w);
13             NodoC e = b.getP();
14             while (e.getSig() != null)
15                 e = e.getSig();
16             e.setSig(w.getSig());
17             w.setSig(b.getP());
18         }
19         w = w.getSig();
20     }
21 }
```

Ejercicios propuestos

- Adicionar k nuevos colegios antes del i -esimo
- Adicionar k nuevos colegios después del colegio con nombre x
- Verificar si existe algún colegio con mas de x estudiantes en el turno “*nocturno*”

Sean 2 listas:

- Insertar solo el primer nodo de la segunda lista después del i -esimo de la primera lista
- Insertar los dos primeros nodos de la segunda lista después del i -esimo de la primera lista
- Insertar los k -primeros nodos de la segunda lista después del i -esimo de la primera lista
- Cuantos nombres de colegio de la primera lista estan en la segunda lista
- En que lista se encuentra el Colegio cx (objeto)

No es necesario que entregue estos ejercicios, solo es para que usted practique