

## TIEMPOS DE EJECUCION DE LAS TRES COLECCIONES

```
/**
 * Vector:
 *   add es O(1)
 *   remove es O(n)
 *   set es O(1)
 *   get es O(1)
 *
 * ArrayList:
 *   add es O(1)
 *   remove es O(n)
 *   set es O(1)
 *   get es O(1)
 *
 * LinkedList:
 *   add es O(1)
 *   remove es O(n)
 *   set es O(n)
 *   get es O(n)
 *
 * Con el iterator:
 *   remove es O(1)
 */
import java.util.ArrayList;
import java.util.LinkedList;
import java.util.ListIterator;
import java.util.Vector;
public class ComparacionEntreArreglos {
    public static void main(String[] args) {
        Vector<Integer> vector = new Vector<Integer>();
        ArrayList<Integer> arreglo = new ArrayList<Integer>();
        LinkedList<Integer> lista = new LinkedList<Integer>();
        LinkedList<Integer> lista1 = new LinkedList<Integer>();
        int limite1 = 100000, limite2 = 9999, limite3 = 10000;

        // Vector add
        long inicio = System.nanoTime();
        for (int i = 0; i < limite1; i++) {
            vector.add(i);
        }
        long fin = System.nanoTime();
        long duracion = fin - inicio;
        System.out.printf("Vector add:           %,20d %n", duracion);

        // ArrayList add
        inicio = System.nanoTime();
        for (int i = 0; i < limite1; i++) {
            arreglo.add(i);
        }
        fin = System.nanoTime();
        duracion = fin - inicio;
        System.out.printf("ArrayList add:       %,20d %n", duracion);

        // LinkedList add
        inicio = System.nanoTime();
        for (int i = 0; i < limite1; i++) {
            lista.add(i);
        }
    }
}
```

```
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("LinkedList add:    %,20d %n", duracion);

System.out.println("-----");

// Vector remove
inicio = System.nanoTime();
for (int i = limite2; i >= 0; i--) {
    vector.remove(i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("Vector remove:    %,20d %n", duracion);

// ArrayList remove
inicio = System.nanoTime();
for (int i = limite2; i >= 0; i--) {
    arreglo.remove(i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("ArrayList remove: %,20d %n", duracion);

// LinkedList remove
inicio = System.nanoTime();
for (int i = limite2; i >= 0; i--) {
    lista.remove(i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("LinkedList remove: %,20d %n", duracion);

// LinkedList remove
for (int i = 0; i < limite1; i++) {
    lista1.add(i);
}
ListIterator<Integer> it = lista1.listIterator(limite1);
inicio = System.nanoTime();
for (int i = limite2; i >= 0; i--) {
    it.previous();
    it.remove();
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("LinkedList remove (it): %,15d %n", duracion);

System.out.println("-----");

// Vector set
inicio = System.nanoTime();
for (int i = 0; i < limite3; i++) {
    vector.set(i,i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("Vector set:          %,20d %n", duracion);
```

```
// ArrayList add
inicio = System.nanoTime();
for (int i = 0; i < limite3; i++) {
    arreglo.set(i,i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("ArrayList set:      %,20d %n", duracion);

// LinkedList add
inicio = System.nanoTime();
for (int i = 0; i < limite3; i++) {
    lista.set(i,i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("LinkedList set:    %,20d %n", duracion);

System.out.println("-----");

// Vector get
inicio = System.nanoTime();
for (int i = 0; i < limite3; i++) {
    vector.get(i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("Vector get:          %,20d %n", duracion);

// ArrayList get
inicio = System.nanoTime();
for (int i = 0; i < limite3; i++) {
    arreglo.get(i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("ArrayList get:      %,20d %n", duracion);

// LinkedList get
inicio = System.nanoTime();
for (int i = 0; i < limite3; i++) {
    lista.get(i);
}
fin = System.nanoTime();
duracion = fin - inicio;
System.out.printf("LinkedList get:     %,20d %n", duracion);
}
}
```