

Apellidos: \_\_\_\_\_ Nombre: \_\_\_\_\_ Grupo: A

## JBenchOO

1. El objetivo general de este examen es convertir un código procedural que se suministra a una solución orientada a objetos.

El programa facilitado *App.java* calcula el tiempo de ejecución de distintas operaciones realizadas sobre un *ArrayList* mediante una aproximación procedural (de forma imperativa vamos indicando las instrucciones que queremos que ejecute el ordenador a través de procedimientos).

Como se puede ver en el código, el diseño del programa no sigue el paradigma orientado a objetos aprendido en la asignatura y se detectan partes de código duplicadas, las cuales se pueden mejorar.

Se observa también que, para cambiar de colección, hay que modificar dicho código mediante comentarios.

La solución propuesta debe cumplir con los siguientes objetivos:

- Aplicar los principios vistos en clase del paradigma orientado a objetos para mejorar el diseño de este programa. Mediante este nuevo enfoque se intentará reutilizar el mayor número de líneas de código posibles. Como se puede ver en la solución suministrada, existe mucho código repetido.
- Programar la solución para que pueda calcular el rendimiento de las estructuras List vistas en clase (*ArrayList* y *LinkedList*). La solución deberá alternar entre ambas estructuras en tiempo de ejecución.
- Hay que implementar adicionalmente la clase *Util*, no suministrada, pero referenciada en el código para proporcionar la funcionalidad que se solicita:
  - *Util.ITERACIONES*: número de iteraciones, por ejemplo, 1000.
  - *Util.getRandom(SIZE)*: devuelve un número entero entre 0 y *SIZE-1*.

### RECOMENDACIÓN:

1. Implementar la clase *Útil* y los elementos que se piden.
2. Probar que el código suministrado es 100% funcional con esa clase y que genera una salida que se corresponde con los tiempos de ejecución para cada operación ejecutada. Los tiempos mostrados no serán representativos, es decir, nos dará igual que muestre 1 que 100.

Herramientas que se podrán utilizar en el examen:

- Editor de texto.
- Compilador.
- JShell: si alguien desea ejecutar código Java de una forma rápida.
- No se podrá consultar la API.

**No se podrá utilizar ningún IDE (Visual Studio Code, IntelliJ, Eclipse, etc.)**

Se pide:

- Implementar todas las clases para dar respuesta al enunciado.
- Mejorar el diseño del programa suministrado. El código entregado, no podrá tener errores de compilación ni de ejecución.
- Seguir la estructura de paquetes: `exámenes.inter.dominio` (objetos) y `exámenes.inter.ui` (App).
- Utilizar **enums**.

## API

Package [java.lang](#)

### Class Math

static double [random](#)()

Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.