

A.E.D. Individual 1

Guillermo Román

guillermo.roman@upm.es

Lars-Åke Fredlund

lfredlund@fi.upm.es

Manuel Carro

mcarro@fi.upm.es

Julio García

juliomanuel.garcia@upm.es

Tonghong Li

tonghong@fi.upm.es

Normas

- Fechas de entrega y nota máxima alcanzable:

Hasta el Martes 23 de septiembre, 16:00 horas	0
Hasta el Miércoles 24 de septiembre, 16:00 horas	20 %
Hasta el Jueves 25 de septiembre, 16:00 horas	40 %

Después la puntuación máxima será 0
- Se comprobará plagio y se actuará sobre los detectados.
- Usad las horas de tutoría para preguntar sobre programación – son oportunidades excelentes para aprender.

Entrega

- Todos los ejercicios de laboratorio se deben entregar a través de
`http://deliverit.fi.upm.es`
- El/los fichero(s) que hay que subir es/son `Utils.java`.
- La clase debe estar en el paquete `aed.loops` .
- La documentación de la API de `aedlib.jar` está disponible en
`http://costa.ls.fi.upm.es/teaching/aed/docs/aedlib/`

Configuración previa

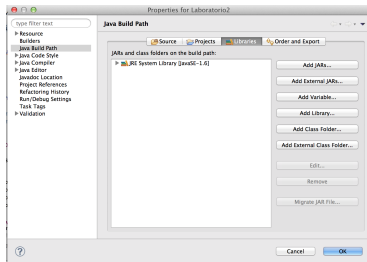
- Arrancad Eclipse
- Si trabajáis en portátil, podéis utilizar cualquier versión reciente de Eclipse. Es suficiente con que instaléis la *Eclipse IDE for Java Developers*.
- Cambiad a “Java Perspective”.
- Debéis tener instalado al menos Java JDK 8.
- Cread un proyecto Java llamado aed:
 - ▶ Seleccionad separación de directorios de fuentes y binarios.
 - ▶ **No debéis elegir la opción de crear el fichero** module-info.java
- Cread un *package* aed.loops en el proyecto aed, dentro de src
- Moodle → AED → Practicas → Individual 1 → Individual1.zip; descomprimidlo
- Contenido de Individual1.zip:
 - ▶ Utils.java, TesterInd1

Configuración previa

- Importad al paquete `aed.loops` los fuentes que habéis descargado (`Utils.java`, `TesterInd1`)
- Añadid al proyecto `aed` la librería `aedlib.jar` que tenéis en Moodle (en Laboratorios).

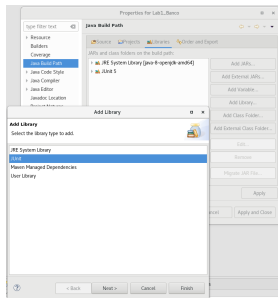
Para ello:

- Project → Properties → Java Build Path. Se abrirá una ventana como la de la izquierda
- Usad la opción “Add External JARs...”
- Si vuestra instalación distingue `ModulePath` y `ClassPath`, instalad en `ClassPath`



Configuración previa

- Añadid al proyecto aed la librería JUnit 5



Para ello:

- Project → Properties → Java Build Path. Se abrirá una ventana como la de la izquierda;
 - Usad la opción “Add Library...” → Seleccionad “JUnit” → Seleccionad “JUnit 5”
 - Si vuestra instalacion distingue ModulePath y ClassPath, instalad en ClassPath
-
- En la clase TesterInd1 tenéis las pruebas, para ejecutarlas, abrid el fichero TesterInd1, pulsando el botón derecho sobre el editor, seleccionar “Run as...” → “JUnit Test”
 - NOTA: Si al ejecutar, no aparece la vista “JUnit”, podéis incluirla en “Window” → “Show View” → “Java” → “JUnit”

Documentación de la librería aedlib.jar

- La documentación de la API de aedlib.jar está disponible en <http://costa.ls.fi.upm.es/teaching/aed/docs/aedlib/>
- También se puede añadir la documentación de la librería a Eclipse (*no es obligatorio*):
 - ▶ En el “Package Explorer”: “Referenced Libraries” → aedlib.jar y elige la opción “Properties”. Se abre una ventana donde se puede elegir “Javadoc Location” y ahí se pone como “javadoc location path:”

<http://costa.ls.fi.upm.es/teaching/aed/docs/aedlib/>
y presionar el botón “Apply and Close”

Tarea: Calcular el número máximo de ocurrencias consecutivas de un elemento en un array

- Se pide implementar el método

```
static int maxNumRepeated(Integer[] array, Integer elem)
```

dentro la clase Utils que recibe un *array* de enteros *array* y un *Integer elem*, y devuelve el número máximo de ocurrencias consecutivas de *elem* en *array*.

- Ejemplos:

<code>maxNumRepeated([], null)</code>	<code>--></code>	<code>IllegalArgumentException</code> lanzado
<code>maxNumRepeated([], 4)</code>	<code>--></code>	<code>0</code> // array vacío
<code>maxNumRepeated([1], 4)</code>	<code>--></code>	<code>0</code> // 4 no ocurre en [1]
<code>maxNumRepeated([1, 4, 3], 4)</code>	<code>--></code>	<code>1</code>
<code>maxNumRepeated([1, 4, null, 4], 4)</code>	<code>--></code>	<code>1</code> // debido a null
<code>maxNumRepeated([1, 4, 3, 4, 4, 3], 4)</code>	<code>--></code>	<code>2</code> // debido a 4,4 en el array
<code>maxNumRepeated([1, 4, 4, 4, 3], 4)</code>	<code>--></code>	<code>3</code> // debido a 4,4,4 en el array
<code>maxNumRepeated([1, 4, 3, 4, 3, 4], 4)</code>	<code>--></code>	<code>1</code> // 4 solo aparece aislado

Notas importantes

- El parametro array nunca va tener el valor `null`.
- El array podria contener elementos `null`.
- El valor de `elem` podria ser `null`, en cuyo caso el método debería lanzar la excepción `IllegalArgumentException`.
- No se debe modificar la estructura de datos recibida como parámetro.
- El proyecto debe compilar sin errores y debe cumplirse la especificación de los métodos a completar.
- Debe ejecutar `TesterInd1` correctamente sin mensajes de error
- **Nota:** un test sin mensajes de error **no** significa que el método sea correcto (es decir, que funcione bien para cualquier posible entrada).
- Todos los ejercicios se comprueban manualmente antes de dar la nota final.