



**Estrategias de Programación
y Estructuras de Datos
Práctica curso 2021-2022**

**Instrucciones para la batería de
pruebas de los estudiantes**

Procedimiento para ejecutar la batería de pruebas

La batería de pruebas de este año consta de dos scripts de ejecución, uno para Windows y otro Linux/Mac, y un conjunto de archivos y carpetas para la configuración de los mismos, que se encuentran recogidos en la carpeta “*juego_de_pruebas_2022*”. En concreto, el contenido del archivo **Bateria_Pruebas_Estudiantes.zip** es:

- **Compile_Run_Test_2022_EST_V.00.bat**: script ejecutable en Windows que aplica las restricciones, compila, ejecuta y evalúa una práctica contra la batería de pruebas proporcionada por el equipo docente.
- **Compile_Run_Test_2022_EST_V.00.sh**: script ejecutable en Linux/Mac que aplica las restricciones, compila, ejecuta y evalúa una práctica contra la batería de pruebas proporcionada por el equipo docente.
- **Juego_de_pruebas_2022**: carpeta con los distintos elementos de configuración de los scripts.
 - *lib*: carpeta que incluye los .jar necesarios para aplicar las restricciones, compilar, ejecutar y evaluar la práctica.
 - **Comparator.jar**: librería que permite evaluar la práctica de acuerdo a la salida obtenida y la salida esperada.
 - **TAD_modified.jar**: TAD modificado para compilar y ejecutar la práctica de acuerdo a las restricciones de la práctica.
 - **BasicTest.jar**: librería que permite evaluar pruebas básicas sobre la práctica de los estudiantes.
 - **Src**: carpeta que incluye varias clases de código que permiten realizar las distintas pruebas de la batería del equipo docente.
 - *pruebas*: carpeta que contiene los archivos de entrada para la ejecución de la batería de pruebas.
 - **JuegoPruebas_Estudiantes_18_Basic_2.dat**: archivo de entrada que contiene la ejecución del sistema en un escenario de prueba básica. Cada línea del archivo contiene una instrucción de las permitidas en la práctica.
 - **JuegoPruebas_Estudiantes_1000_100.dat**: archivo de entrada que contiene la ejecución del sistema en un escenario de pocas instrucciones y pocos índices. Cada línea del archivo contiene una instrucción de las permitidas en la práctica.
 - **JuegoPruebas_Estudiantes_15000_100.dat**: archivo de entrada igual que el anterior, pero diseñado para un escenario de muchas instrucciones y pocos índices.
 - **JuegoPruebas_Estudiantes_30000_100000.dat**: archivo de entrada igual que el anterior, pero diseñado para un escenario de muchas instrucciones y muchos índices.
 - *salida*: carpeta donde se almacenan las salidas de las ejecuciones de la batería de pruebas, así como los archivos de las salidas esperadas con los que comparar.

- **Salida_18_Basic_2.dat:** archivo de salida que contiene la salida esperada para la ejecución de la batería del escenario de pruebas básicas.
- **Salida_1000_100.dat:** archivo de salida que contiene la salida esperada para la ejecución de la batería del escenario pocas instrucciones y pocos índices.
- **Salida_15000_100.dat:** archivo de salida que contiene la salida esperada para la ejecución de la batería del escenario muchas instrucciones y pocos índices.
- **Salida_30000_100000.dat:** archivo de salida que contiene la salida esperada para la ejecución de la batería del escenario muchas instrucciones y muchos índices.

Para que la batería de pruebas funcione es importante mencionar que es necesario tener **instalado el JDK de Java**, por lo que no es suficiente con tener el JRE.

Para ejecutar la batería de pruebas simplemente hay que descomprimir el contenido del archivo **Bateria_Pruebas_Estudiantes.zip** en la carpeta principal del proyecto Eclipse (en general, todos los entornos de desarrollo en Java siguen un esquema muy similar al de Eclipse), o lo que es lo mismo, donde se encuentre la carpeta *src* de la práctica.

Es importante mencionar que la batería de pruebas está diseñada para la estructura y nombres de clases propuestos en la práctica y en las estructuras de datos, por lo que, si se ha modificado en algún aspecto esta estructura o nombres, la batería de pruebas puede no funcionar.

Una vez copiados los archivos simplemente se deberá ejecutar el script .bat, si se está en un entorno Windows, o .sh, si se está en un entorno Linux/Mac. Para ello, simplemente hay que ejecutar desde la línea de comandos:

```
> Compile_Run_Test_2022_EST_V.00.bat
```

en entorno Windows.

```
> ./Compile_Run_Test_2022_EST_V.00.sh
```

en entorno Linux/Mac.

Es importante mencionar que los archivos .bat y .sh pueden no funcionar si se ejecutan directamente desde el entorno de desarrollo Eclipse, o cualquier otro entorno de desarrollo, por lo que es recomendable ejecutarlos mediante consola.

Un aspecto importante es que el script comprueba si el sistema operativo tiene o no definida la variable de entorno `JAVA_HOME`. En caso afirmativo, el script se ejecuta sin problemas. En caso de no tener asignada la variable `JAVA_HOME`, será necesario editar el script e incluir la ruta del JDK en el sistema. Para ello, debemos modificar la variable que apunta al `JAVA_HOME` de los scripts tal y como se muestra a continuación. En el caso del entorno Windows:

```
set JAVA_HOME_JDK=""
```

por

```
set JAVA_HOME_JDK=" C:\Program Files\Java\jdk1.8.0_101"
```

Es importante mencionar que el directorio puede variar dependiendo de la versión de Windows, de la versión de Java y del proceso de instalación del JDK. En caso de un entorno Linux/Mac sería:

```
JAVA_HOME_JDK=
```

por

```
JAVA_HOME_JDK=/usr/lib/jvm/default-java
```

De igual manera, la ruta del JDK depende del sistema Linux/Mac, del JDK instalado y del proceso de instalación.

Una vez configurado el entorno Java en el script, éste irá mostrando los diferentes pasos que va realizando en cada ejecución. Es importante mencionar que en cada ejecución crea una carpeta temporal, *tmp*, dentro de la carpeta *juego_de_pruebas_2022*, donde se realizan todas las operaciones de compilación y ejecución.