

Curso 2021 - 2022

Práctica Final – Guerra de Naves

Objetivos

- Practicar la creación y gestión de hilos en UNIX.
- Sincronizar el funcionamiento de hilos en UNIX.
- Practicar la gestión y control de acceso a memoria compartida mediante semáforos.
- Implementación de listas enlazadas con reserva dinámica de memoria.

Descripción general

El objetivo de la práctica es diseñar un sistema concurrente para determinar qué nave ha ganado la batalla.

El sistema deberá procesar una sucesión de caracteres que se proporcionan en un fichero de texto. Cada carácter del fichero se corresponderá con un token que tendrá una longitud de uno o dos caracteres. Los tokens posibles son:

- Carácter ***, indica que la nave ha sido acertada por un disparo.
- Carácter *espacio*, indica que la nave no ha sido acertada por un disparo.
- Carácter *b*, indica que la nave ha conseguido un botiquín. El token de tipo botiquín llevará detrás un número que indicará el número de botiquines (será un número entre 1 y 3). Por ejemplo, *b2* indica que hay dos botiquines.

El programa principal recibirá cuatro parámetros:

- Nombre de fichero de entrada del que se leerán los tokens.
- Nombre de fichero de salida donde se escribirán los resultados.
- Tamaño del buffer circular.
- Número de naves.

El sistema deberá constar de una serie de hilos que deben lanzarse al comienzo del proceso de forma concurrente:

- **Disparador:** Se encargará de:
 - Abrir el archivo de entrada proporcionado por el programa principal. Será un archivo de texto.
 - Leer del archivo un número desconocido de *tokens*.
 - De los tokens leídos, sólo se quedará con los que correspondan con un token válido que almacenará en un buffer circular del que leerán las naves de manera concurrente.
 - Una vez procesado todo el fichero escribirá en el fichero de salida el número de tokens válidos e inválidos procesados, así como el total de tokens leídos.

Los alumnos disponen de dos ficheros de ejemplo que puede obtenerse en la ruta: **/home/javramo/tmp/bombardeo.dat** y **/home/javramo/tmp/bombardeolargo.dat**. El primero se usará para desarrollar el programa y el segundo para verificar su funcionamiento. Para facilitar la verificación de resultados del programa se puede consultar el fichero **/home/javramo/tmp/bombardeoDATOS.dat** donde aparecen las estadísticas y distribución de elementos y tokens del fichero **/home/javramo/tmp/bombardeo.dat**

- **Naves:** Las naves se encargarán de leer del buffer circular los tokens que el disparador vaya escribiendo en él. Cada token leído se deberá procesar de la siguiente forma:
 - Si el token es un asterisco *"*"* contará como un impacto y se deberá consumir ese token del buffer circular.
 - Si el token es un espacio en blanco *" "* contará como un intento fallido y se deberá consumir ese token del buffer circular.

- Si el token es un botiquín, la nave acumulará un botiquín, se restará uno al botiquín en el buffer circular del disparador y se dejará en el buffer para que pueda ser usados por otras naves. Si el botiquín llega a cero, se deberá sacar de buffer circular.
 - Cada nave deberá llevar la cuenta de tokens obtenidos, disparos recibidos y fallados, así como el número de botiquines obtenidos.
 - Una vez que se hayan consumido todos los tokens del buffer circular, las naves deberán terminar y guardar en **una lista enlazada los datos obtenidos** (disparos recibidos, fallados y número de botiquines) para que el juez realice el cálculo final.
- **Juez:** recogerá la información de los resultados de las naves en el orden en el que éstas vayan terminando, recorriendo la lista en cuanto tenga resultados de alguna nave y realizará las siguientes tareas:
 - Deberá añadirse al archivo de resultados los datos de cada nave con el formato que se muestra en el siguiente ejemplo. El orden de escritura del archivo será el de terminación de las naves.
 - Cuando termine de procesar los datos de todas las naves, deberá escribir los datos del ganador de la batalla y del subcampeón.
 - Por último, deberá escribir un resumen de la suma de todos los datos almacenados en la lista enlazada.

```

El disparador ha procesado: 85 tokens válidos y 15 tokens inválidos, total: 100
Nave: 1
    Disparos recibidos: 50
    Disparos fallados: 3
    Botiquines obtenidos: 12
    Puntuación: 38

Nave: 0
    Disparos recibidos: 18
    Disparos fallados: 3
    Botiquines obtenidos: 7
    Puntuación: 11

Nave: 3
    Disparos recibidos: 0
    Disparos fallados: 0
    Botiquines obtenidos: 0
    Puntuación: 0

Nave: 2
    Disparos recibidos: 0
    Disparos fallados: 0
    Botiquines obtenidos: 0
    Puntuación: 0

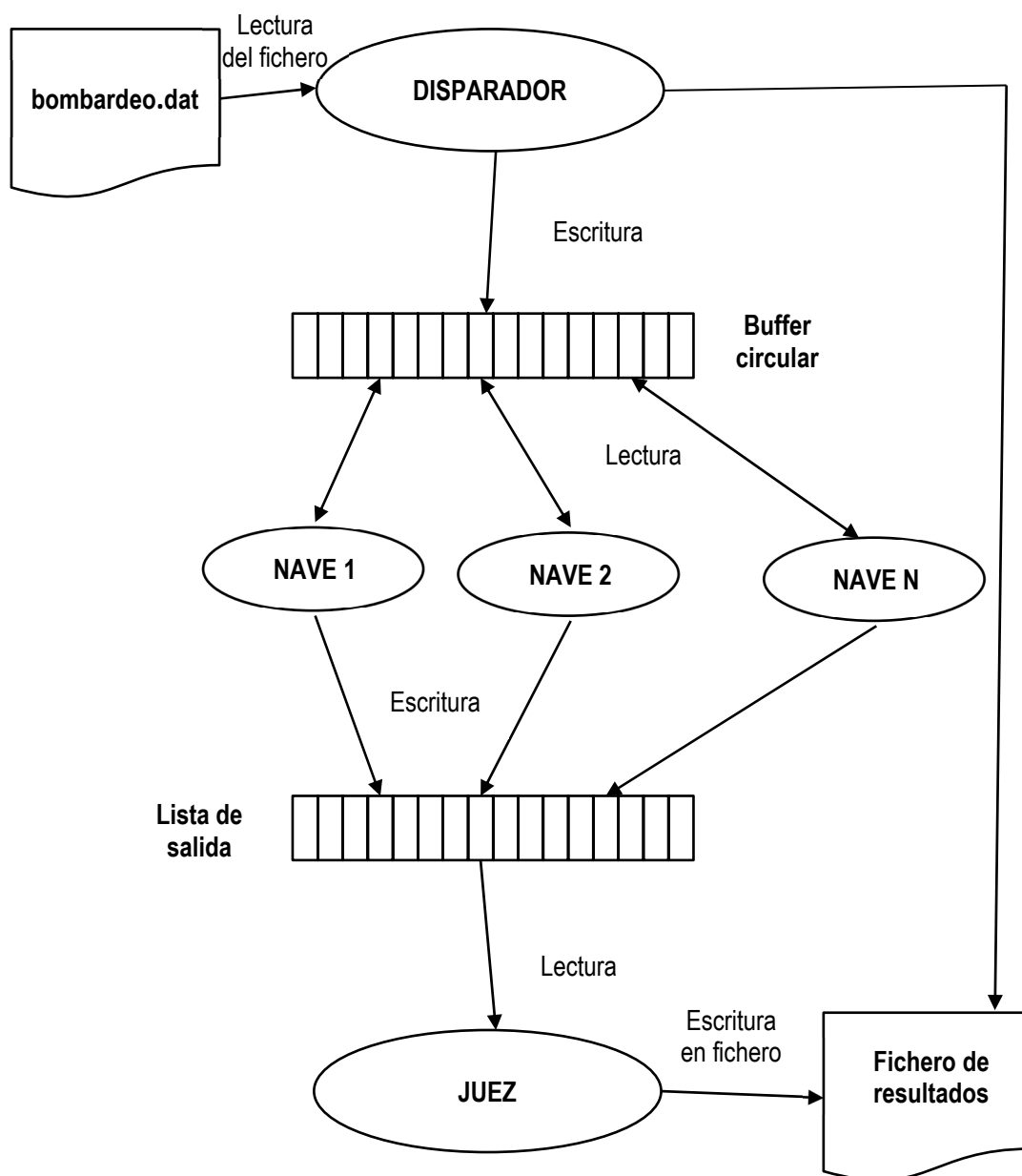
Nave: 4
    Disparos recibidos: 0
    Disparos fallados: 0
    Botiquines obtenidos: 0
    Puntuación: 0

Nave Ganadora: 0
    Disparos recibidos: 18
    Disparos fallados: 3
    Botiquines obtenidos: 7
    Puntuación: 11

Nave Subcampeona: 1
    Disparos recibidos: 50
    Disparos fallados: 3
    Botiquines obtenidos: 12
    Puntuación: 38

===== RESUMEN =====
    Disparos recibidos Totales: 68
    Disparos fallados Totales: 6
    Botiquines obtenidos Totales: 19
    Total de tokens emitidos: 93
  
```

En la siguiente imagen se muestra de manera esquemática los diferentes hilos y recursos de los que consta el sistema.



Desarrollo de la práctica:

La aplicación desarrollada deberá tener la siguiente sintaxis a la hora de ser invocada para su ejecución:

```
./<program> <inputFile> <outputFile> <tamBuffer> <numNaves>
```

Seguimiento y entrega parcial

La práctica se debe realizar tanto en las sesiones de prácticas como de manera autónoma. Cada grupo de prácticas debe dejar constancia al terminar cada sesión de prácticas del trabajo que lleva realizado y de cómo va a afrontar los próximos pasos.

Entrega 0: Constará del disparador y de una única nave. No será necesario implementar el juez ni el fichero de resultados. Esta entrega debe subirse en el Moodle de la asignatura el día 2 de diciembre de 2002 antes de las 23:59. Se entregará lo que se haya hecho hasta ese momento independientemente de si se han conseguido todos los requisitos de esta entrega.

Entrega 1: Práctica completa en la que la lista enlazada puede ser sustituida por un vector de tamaño igual al número de naves. Esta entrega debe subirse en el Moodle de la asignatura el día 16 de diciembre de 2002 antes de las 23:59. Se entregará lo que se haya hecho hasta ese momento independientemente de si se han conseguido todos los requisitos de esta entrega.

Entrega 2: Práctica completa. El día 22 de diciembre de 2020 antes de las 23:59 se deberá realizar la entrega definitiva de la práctica.

El cumplimiento de las entregas en plazo se reflejará en el 10% de la nota total de la práctica.

Cada grupo defenderá la práctica ante el profesor, donde se evaluará entre otras cosas que sea coherente con el trabajo realizado a lo largo de las sesiones de prácticas previas. Entre los días 11 de Enero de 2021 y el 29 de Enero de 2021 se deberá concertar una hora para realizar la defensa de la práctica según se acuerde con el profesor.

Condiciones que deben cumplir las entregas:

- La entrega se realizará mediante la web de la asignatura en la plataforma Moodle aulas.inf.uva.es
- Se entregará el código fuente del programa en lenguaje C realizado.
- El código debe estar desarrollado en C estándar y compilar en la máquina jair.lab.inf.uva.es.
- El programa debe ser concurrente.
- Para una mejor identificación, el nombre del programa indicará el grupo de laboratorio y primer apellido de cada miembro de la pareja; será algo así como: X2_GonzalezMerino_ParteX.c. IMPORTANTE: si los apellidos tienen acento, no se pondrán en el nombre del programa.
- En las primeras líneas de cada fichero fuente enviado, debe aparecer como comentario el nombre, dos apellidos y DNI de cada uno de los autores.
- Se podrá incluir cualquier información adicional que se considere que puede ser útil para la evaluación de la práctica.
- Se valorará la claridad del código y la inclusión de comentarios aclaratorios.