

Fundamentos de la programación – Curso 2012-2013

Corrección de la práctica 1

(Documento desarrollado por Pablo Moreno Ger, profesor del grupo E)

Guión de pruebas

1. Vuelve a leer todo el enunciado y en las explicaciones adicionales de la práctica. Todo lo que pone en el enunciado es obligatorio.
2. Comienza la ejecución con los archivos de reglas y registros incluidos en la demo de la práctica.
3. Probar a entrar con el usuario “luis”.
 - Probar con la contraseña “XXX” (no debería aceptarla)
 - Introducir otras dos contraseñas incorrectas. El juego debería decir que no es un usuario autorizado y terminar la ejecución.
4. Lanzar otra vez el programa, con un nuevo nombre de usuario (spaceball). Debería pedir que introduzca la contraseña:
 - Pon la contraseña 12345.
5. Termina la ejecución sin jugar ninguna partida.
6. Verifica que el nuevo usuario se ha añadido al final del archivo de registro, seguido de su contraseña y, por último, el centinela.
7. Arranca de nuevo el programa, selecciona el usuario spaceball.
 - Introduce la contraseña 00000 (la rechaza)
 - Introduce la contraseña 1234 (la rechaza)
 - Introduce la contraseña 12345 (la acepta y empezamos a jugar).
8. Prueba el apartado de mostrar el listado de reglas.
 - Pruébalo dos veces seguidas para asegurarte de que no estás teniendo problemas con la apertura y cierre del archivo.
 - Prueba a mostrar el listado de reglas si el archivo no existe.

9. Juega varias partidas seguidas y comprueba que se generan bien los resultados:

- Prueba al menos una vez todos los posibles empates
- Prueba al menos una vez todas las posibles victorias del humano contra cada selección de la máquina (ganar vez contra piedra, ganar contra papel, ...).
- Prueba al menos una vez todas las posibles derrotas del humano contra cada selección de la máquina (perder contra piedra, perder contra papel, ...).
- Para facilitar las pruebas puedes poner al principio del programa, una sola vez, la semilla para el generador aleatorio. Si pones una semilla fija, siempre saldrán los mismos datos para la máquina. Esto facilita mucho el proceso de pruebas anterior. Por ejemplo, si ponemos como semilla 42 (`srand(42);`) la máquina genera las siguientes selecciones en este orden:

```
piedra >> piedra >> spock >> papel >> lagarto >> spock >>
papel >> spock >> spock >> spock >> spock >> piedra >>
papel >> spock >> papel >> piedra >> lagarto >> tijeras >>
papel >> papel >> tijeras >> piedra >> papel >> tijeras
```

10. Cuando estés satisfecho con que las partidas se resuelven bien, prueba las validaciones de datos:

- Prueba a introducir una contraseña vacía. ¿Qué sucede? Es aceptable casi cualquier comportamiento que no suponga que el programa o los archivos se corrompan.
- En el menú principal, prueba a introducir como opción un -1 o un 42.
- Al leer la jugada del usuario, prueba a escribir “tigera”.
- Al leer la jugada del usuario, prueba a escribir “spock” y “Spock”. Es aceptable que una de las dos la rechace.

El programa debe validar las entradas, pero puede ignorar los problemas de tipos.

Por ejemplo: Si hay que pedir un número, deberá comprobar que sea un número perteneciente al rango de valores válidos. Si hay que pedir una cadena, que sea una cadena válida. Pero si hay que pedir un número y el usuario escribe una cadena (lo cual rompe el programa) no es necesario que arregles nada.