



Práctica de curso 2022-23

El objetivo de la práctica es elaborar un programa Java que implemente el juego “Wordle” actuando como parte activa, es decir, como habitualmente hace el jugador humano.

Juego Wordle

Wordle es un juego creado por Josh Wardle y publicado en octubre de 2021, que ha tenido un enorme éxito en poco tiempo. Combina ideas de otros juegos como “El ahorcado” o “Mastermind”. Se juega desde una página web (<https://www.nytimes.com/games/wordle/index.html>), con la particularidad de que cada jugador solamente puede hacerlo una vez al día. Existen versiones en otros idiomas (como en español: <https://lapalabradeldia.com/>)

El juego consiste en adivinar una palabra oculta de 5 letras en 6 intentos o menos. En cada intento, el jugador propone una palabra y recibe pistas acerca de las letras de su propuesta que son correctas, que están mal situadas, o que no están en la palabra oculta, y puede usar esta información para seleccionar nuevos candidatos para sus siguientes intentos. La palabra oculta pertenece a un diccionario de palabras consideradas frecuentes en el idioma en cuestión, mientras que los intentos del jugador deben ser palabras también de un diccionario, pero más extenso.

Las pistas se ofrecen mostrando la palabra del intento con el fondo de cada letra coloreado, con las siguientes reglas:

- verde significa que la letra está en la palabra y en la posición correcta.
- naranja significa que la letra está presente en la palabra pero no en la posición correcta.
- gris significa que la letra no está en la palabra.

La palabra oculta puede tener letras repetidas. En ese caso, las pistas son independientes para cada letra y la primera regla tiene prioridad. Por lo tanto, una letra repetida en la propuesta del jugador podrá tener más de un color de fondo (verde o amarillo) solo si aparece repetida en la oculta, y aparecerá con color tantas veces como aparezca en la oculta.

Se muestra un ejemplo de un juego. La palabra oculta es “MARCA”. En la primera propuesta no hay letras acertadas. En la tercera hay 2 letras acertadas pero en posiciones incorrectas. En la segunda y quinta puede verse el comportamiento con letras repetidas.



Figura 1: partida ejemplo

El programa:

En esta práctica, se pide un programa que actúe como **parte activa** del juego. Es decir: **es el usuario el que piensa una palabra y el programa el que tratará de adivinarla**. En lugar de un código de colores, el usuario responderá a cada propuesta del programa con una cadena de dígitos que califica a propuesta, representando el 0 el color gris, el 1 el naranja y el 2 el verde.

Programa	Usuario
Juguemos a Wordle	
Piensa una palabra ...	
Y dime si la acierto:	
ESTIO	00000
AVALA	10002
PRADO	01100
RASGO	12000
HABRA	02012
FARRA	02202
¡HE PERDIDO!	
¿Cuál era la palabra oculta?	
MARCA	
¿La puedo añadir a mi diccionario? Si	
Gracias.	
¿Otra partida?	SI
...	

Figura 2: Ejemplo de ejecución del programa

Para ello, el programa dispondrá de un diccionario de palabras de 5 letras y tendrá prevista la posibilidad de ampliarlo con unas cuantas palabras más. Se trabajará solamente en mayúsculas. No se usarán eñes, y las palabras se tratarán sin tildes, aunque su grafía correcta las tenga (en el ejemplo anterior, la palabra existente es “habrá”, pero en el programa se considera “HABRA”).

No es necesario que el programa utilice ninguna estrategia, más allá de buscar en cada intento una palabra de su diccionario que cumpla las condiciones deducidas de los intentos anteriores. Si no quedan palabras cumpliéndolas, el programa puede jugar con una palabra cualquiera, hasta completar los 6 intentos.

En el caso de que el programa pierda, le preguntará al usuario cuál era la palabra pensada. Si esta palabra no está en el diccionario del programa, y aún le queda espacio en él, preguntará al usuario si quiere añadirla al diccionario para siguientes partidas, y actuará en consecuencia.

Al terminar una partida (es decir, al haber acertado la palabra o agotar los 6 intentos permitidos etc.), se le preguntará al usuario si quiere jugar más partidas.

Versión básica (máximo 7 puntos de 10):

El juego tal y como se ha descrito, suponiendo que el usuario no hace trampas y responde con calificaciones coherentes. Sí que habrá que validar las entradas de usuario según las características esperadas en cada caso (cadena de ceros, unos y doses, respuestas “sí”, “no”, etc).

Versión completa (máximo 10 puntos de 10):

El diccionario se guardará en un fichero, de forma que las palabras aprendidas en una ejecución del programa se conocerán en las restantes ejecuciones. Siempre se trabajará con un máximo de 100 palabras, con un diccionario inicial de al menos 20.

También se guardará en un fichero el número de partidas jugadas, y el número de partidas ganadas respectivamente por el usuario y por la máquina, y se informará de este estado al empezar y terminar la ejecución del programa.

El programa, una vez terminada cada partida perdida por el programa (y conocida por lo tanto la palabra oculta), detectará además si el usuario se ha equivocado o ha hecho trampa en las cadenas de ceros, unos y doses que proporcionó, confrontándolas¹ con la palabra oculta; si alguna no es correcta, el programa no se considerará perdedor.

Versión especial

Cualquier otra mejora o ampliación del juego deberá ser comentada con el profesor de laboratorio, que podrá aumentar hasta en 2 puntos la nota de la práctica.

Entrega de la práctica

La práctica debe realizarse por parejas del mismo grupo de laboratorio. Cualquier otra situación deberá ser acordada con los profesores de prácticas.

Será calificada con arreglo a los criterios de calidad explicados en la asignatura y el documento “Malas prácticas en programación”, y considerando el acto de defensa.

Solo un alumno de cada pareja debe entregar la práctica. En todos los documentos entregados deben aparecer los nombres de los alumnos que la han realizado, junto con sus grupos de teoría y laboratorio. Ambos alumnos se hacen responsables de la entrega.

Se debe entregar, en el entorno Moodle:

- El **código** del programa en un único archivo .java de nombre elegido por los alumnos
 - Debe estar codificado en utf-8 y no contener línea “package”
 - Debe ser compilable desde la línea de órdenes con la orden “javac *nombre.java*” y poder ser ejecutado con la orden “java *nombre*”
- La **documentación** del programa en un archivo pdf, de no más de 6 páginas, en la que debe describirse:
 - qué versión se ha realizado y la estrategia general de resolución
 - la estructura de uso de módulos del programa (“métodos”, “subprogramas”), así como la especificación de cada uno de ellos (objetivo, precondiciones, entradas, salidas ...)
 - las estructuras de datos relevantes utilizadas en el programa, con una explicación de su contenido y significado en el programa
 - el diccionario inicial

Esta documentación no debe contener el código del programa

Defensa

El profesor de prácticas podrá convocar a los alumnos un acto de “defensa de la práctica”. En ella, el profesor hará preguntas a los alumnos, relacionadas con la práctica entregada y su elaboración, y puede pedir la realización de alguna modificación o cambio en el enunciado. El resultado de la defensa afectará a la calificación obtenida por cada alumno en la práctica.

¹ Para este proceso se puede usar el código proporcionado en la página de la asignatura “Un código de ayuda para la práctica”