

## Trabajo Práctico Especial

### Objetivo

El objetivo del presente trabajo es implementar un juego en lenguaje Java, con interfaz visual, aplicando los conceptos sobre diseño orientado a objetos adquiridos en la materia.

### Descripción funcional

La aplicación a implementar es una variante reducida del juego *Desktop Dungeons*<sup>1</sup>. El juego consiste en un tablero por el cual el jugador se puede desplazar, donde hay enemigos con los que éste puede enfrentarse. Tanto el jugador como los enemigos tienen:

- **Salud:** es un número que determina la salud del personaje. Un personaje tiene una cantidad máxima de salud que es con la que comienza el juego. Durante el transcurso del mismo, esta va disminuyendo a medida que se enfrenta con otros personajes. Si este valor llega a 0, el personaje muere.
- **Fuerza:** es un número que determina cuánto daño le produce a sus oponentes en un ataque. El oponente pierde tantas unidades de salud como fuerza tiene el personaje que lo ataca.
- **Nivel:** es un número entre entero que da una medida de cuán bueno es un personaje. El nivel determina la cantidad máxima de salud y la fuerza que tiene el mismo. Por ejemplo un jugador de nivel 1, tiene una salud máxima de 10 unidades y una fuerza de 5 unidades. A medida que el jugador sube de nivel, aumenta proporcionalmente su fuerza y su salud máxima,

El jugador además cuenta con **experiencia** que es un valor que registra su grado de evolución en su nivel actual, al completar el máximo de experiencia de un nivel el jugador avanza al nivel siguiente.

El objetivo del juego es vencer a todos los enemigos que aparezcan en el tablero. El jugador inicialmente comienza con 10 unidades de salud (sobre un máximo también de 10) y 5 unidades de fuerza. Los enfrentamientos no se pueden realizar en cualquier orden ya que el jugador debe buscar primero a los enemigos más débiles, para vencerlos y así ganar experiencia, aumentar su nivel y entonces si pelear con enemigo más fuertes.

Inicialmente el tablero comienza mostrando únicamente la celda donde está ubicado el jugador y las ocho aledañas. Las restantes celdas del tablero se van descubriendo a medida que el jugador se desplaza por el mismo, siempre mostrando las ocho celdas que lo rodean.

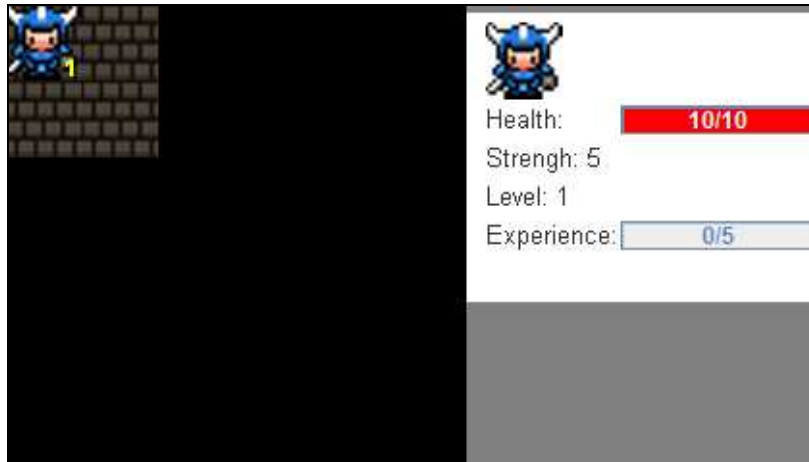
El tablero puede contar con paredes que son elementos que impiden el paso del jugador. Si bien el tablero no necesariamente está rodeado por paredes, no se debe permitir que el jugador salga del mismo.

A la derecha del tablero se muestra el estado del jugador que consta de: imagen, cantidad actual y máxima de unidades de salud, fuerza, nivel y experiencia acumulada sobre la requerida para aumentar su nivel. Sobre la imagen del jugador en el tablero aparece un número que indica su nivel.

En la siguiente imagen se muestra el estado inicial de un tablero de 6 x 6 en donde se pueden observar los atributos del jugador y las celdas descubiertas:

---

<sup>1</sup> <http://www.desktopdungeons.net/media/>



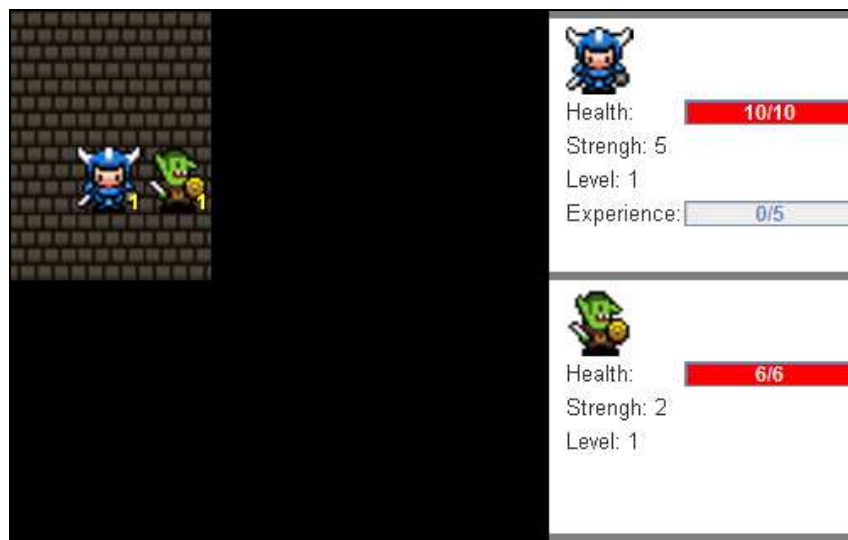
### Enfrentamientos

Para poder vencer a un enemigo es necesario enfrentarse a él. Se produce un enfrentamiento cuando el jugador se encuentra ubicado en una celda adyacente a la de un enemigo e intenta moverse hacia la misma. Cuando esto ocurre, el enemigo ataca en primer lugar al jugador, restándole tantas unidades de salud como fuerza él tenga. Si el jugador sobrevive, ataca al oponente de la misma manera (restándole tantas unidades de salud como fuerza tenga). En caso de que se agote la salud del enemigo, este muere y es reemplazado en el tablero por una mancha de sangre. Además se incrementa la experiencia del jugador tantas unidades como nivel tenga el oponente vencido. Notar que para que el jugador venza al enemigo puede que sea necesario más de un enfrentamiento.

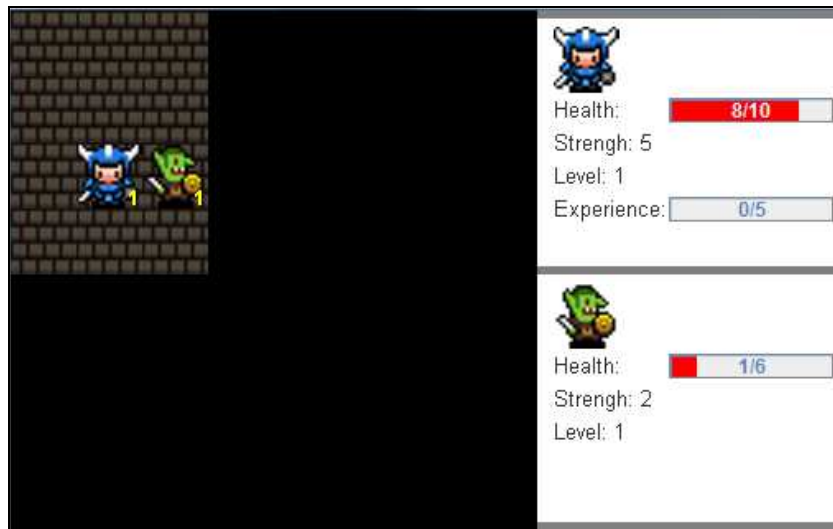
Cuando el usuario posiciona el mouse sobre algún enemigo que se encuentre visible, se deben mostrar las características del enemigo (tipo, salud, fuerza y nivel).

La salud del jugador se recupera cuando este explora celdas aún no descubiertas del tablero. Por cada celda descubierta, se incrementa la salud del jugador tantas unidades como nivel tenga.

En la siguiente imagen se observa al jugador en una celda adyacente a un enemigo. Sobre el margen derecho se observan tanto las características del jugador como del enemigo (esto último, por haber ubicado el mouse encima del enemigo)

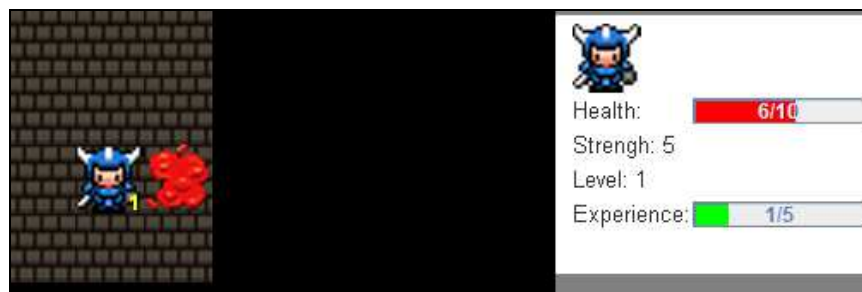


Si el usuario presiona la flecha derecha se produce un enfrentamiento. El enemigo ataca primero al jugador, reduciéndole su salud en 2 unidades. Luego, como el jugador permanece con vida, ataca al oponente quitándole 5 unidades de salud.



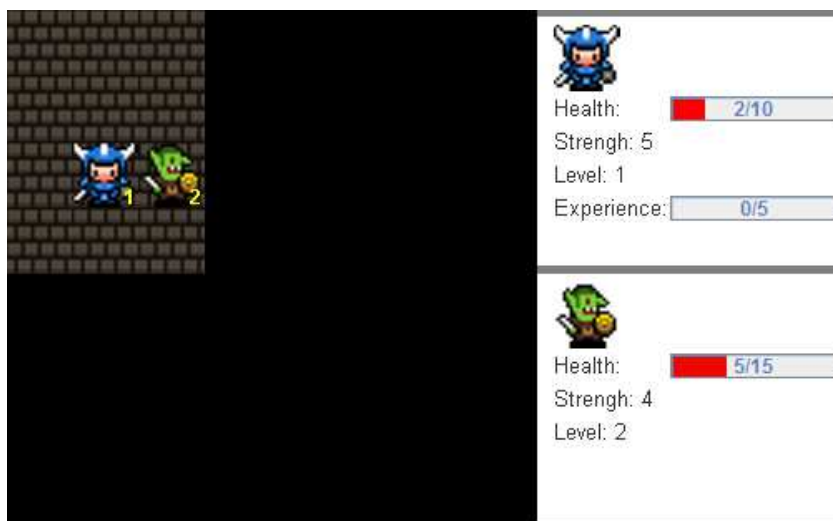
Inmediatamente se observan los cambios en la salud de cada uno. Para que el jugador venza al enemigo será necesario que peleen nuevamente.

Si se vuelve a presionar la flecha derecha, se produce un nuevo enfrentamiento. En primer lugar, el jugador pierde 2 unidades de salud. Luego, al atacar al oponente lo vence debido a que su salud restante es menor a la fuerza del jugador. Cuando esto ocurre, acumula una unidad de experiencia (explicado más adelante en la sección “Niveles”). La imagen del enemigo se reemplaza por una mancha de sangre y a la derecha del tablero no se muestra más la información del enemigo.



Luego de vencer al oponente, si el jugador presiona la flecha derecha, avanza sobre la celda con la mancha de sangre y la misma desaparece.

Lo que sigue es un ejemplo donde el jugador está en desventaja:



Si se presiona la tecla derecha, el enemigo ataca al jugador quitándole 4 unidades de salud (más de las que tiene), en este caso el jugador es el que muere y termina el juego.

### Niveles

El nivel de un personaje determina su salud máxima y su fuerza. Para el jugador que se encuentra en el nivel  $N$ , su salud máxima es  $10*N$  y su fuerza  $5*N$  (más pueden influir los elementos recogidos descriptos en la sección “Elementos”).

Para el caso de los enemigos estos valores dependen tanto de su nivel como de su tipo. Existen 3 tipos de oponentes: Goblin, Golem y Serpiente. Cada uno tiene alguna característica especial.

La salud máxima y la fuerza de un enemigo de nivel  $N$  están dadas por:

$$\text{Salud máxima: } \text{Piso} \{ [ (N + 3)^2 - 10 ] * S \}$$

$$\text{Fuerza: } \text{Piso} [ (N^2 + 5 * N) * 0.5 * F ]$$

donde  $F$  y  $S$  son constantes que dependen del tipo de enemigo según la siguiente tabla:

Enemigo	$F$	$S$	Característica especial
Goblin	0.7	1	Recupera una unidad de vida cada dos movimientos del jugador
Golem	1	1.35	Al morir, en lugar de una mancha de sangre, deja en su celda una espada (con nivel de daño equivalente al nivel a su nivel) que luego puede ser recogida por el jugador
Serpiente	1	1	-

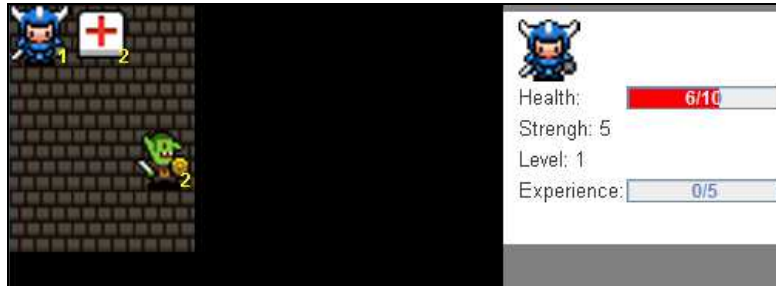
El nivel de los oponentes **no cambia** durante el desarrollo del juego.

A medida que el jugador gana enfrentamientos, acumula unidades de experiencia que le permiten aumentar su nivel. La cantidad de experiencia que acumula está dada por el nivel del enemigo que vence. Cuando un jugador de nivel  $N$  acumula  $5*N$  unidades de experiencia incrementa su nivel (aumentando 10 unidades su salud máxima y 5 su fuerza). Cuando esto ocurre, la cantidad de experiencia vuelve a 0 y la salud toma el valor máximo del nuevo nivel.

## Otros objetos

El tablero además de contener enemigos, paredes y al jugador mismo puede contener otros objetos que el jugador puede recoger a medida que se desplaza.

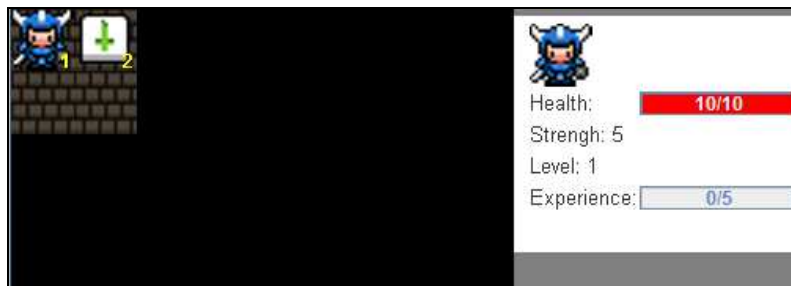
- El **bonus de salud** restaura tantas unidades de salud como éste lo indique. En el siguiente ejemplo, el jugador tiene 6 unidades de salud y se encuentra en una celda adyacente a un bonus de 2 unidades de salud:



Al desplazarse hacia la derecha, levanta el bonus e incrementa 2 unidades su salud. El bonus desaparece del tablero.

Si al sumar el valor de bonus con la salud actual del jugador se excede del máximo, se establece la salud en el máximo valor posible. Si el jugador ya posee la máxima salud posible, consume el bonus de todos modos (sin modificar su salud).

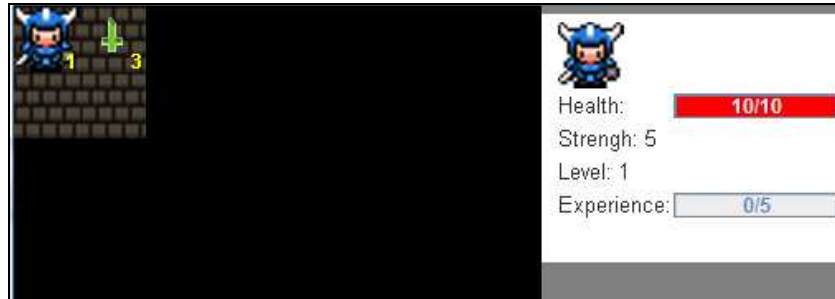
- Existe otro tipo de bonus que incrementa la fuerza tantas unidades como se indica en la figura. En el siguiente ejemplo el jugador se encuentra en una celda adyacente a un **bonus de fuerza 2**. Al desplazarse hacia la derecha, el jugador consume el bonus de fuerza incrementando así esta propiedad en 2 unidades.  
Este bonus de fuerza se pierde al pasar de nivel.



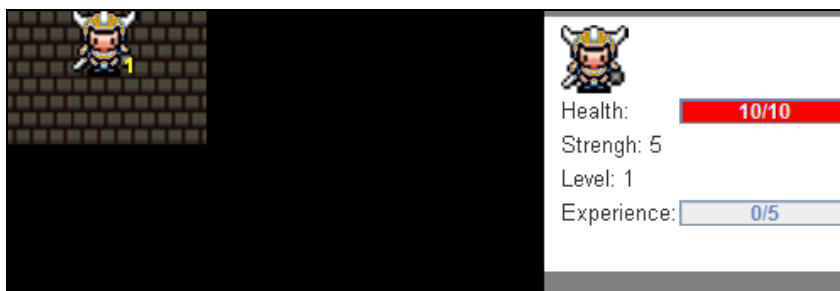
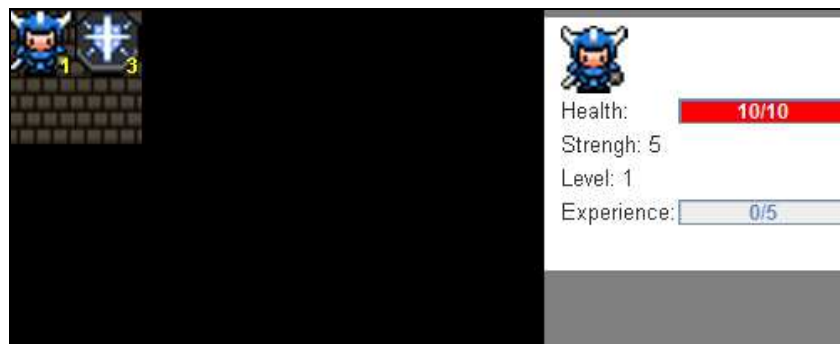
- La **poción de salud** otorga el nivel máximo de salud al jugador. Si el jugador se desplaza sobre ella, se eleva el nivel de salud al máximo y la poción desaparece. Si el jugador ya tiene el valor máximo se consume sin producir efecto.



- Las **espadas** otorgan mayor poder de daño al jugador en forma permanente (aún cuando pase de nivel). Cuando el jugador recoge una espada se convierte en jugador armado. Cuando el jugador está armado, se pueden recoger otras espadas pero las mismas no se acumulan sino que al recoger una se descarta la que tuviere en ese momento. Cada espada incrementa el poder de daño según lo que indique el número debajo de su imagen.



- La **escudo** funciona como defensa. cuando el jugador lo recoge se convierte en jugador con armadura. Como se observa en la sucesión de imágenes cuando el jugador se desplaza sobre el escudo cambia su imagen.



Cuando el jugador tiene armadura, los enemigos reducen su daño en un 30%.

## Consideraciones de implementación

La cátedra proporcionará una versión incompleta del juego descrito, los alumnos deben completar la aplicación ya sea agregando nuevas clases y/o modificando las existentes, de modo que se cumpla la descripción funcional.

Adicionalmente se proporcionarán imágenes para los elementos mencionados. Cada grupo puede optar por utilizar estas imágenes o crear las suyas.

## Grupos

El trabajo deberá ser realizado por grupos de 3 (tres) alumnos. **Cada grupo deberá enviar un correo electrónico a [zabaleta.laura@gmail.com](mailto:zabaleta.laura@gmail.com) informado quiénes son los integrantes.**

## Material y fecha de entrega

Se deberá entregar el proyecto con las nuevas clases y las modificaciones realizadas. La entrega final del trabajo se realizará enviando lo solicitado en un mail a la cuenta **[zabaleta.laura@gmail.com](mailto:zabaleta.laura@gmail.com)**.

La fecha de entrega es el día **viernes 8 de noviembre antes de las 18hs**. Si el trabajo no estuviera aprobado se cuenta con una instancia de recuperación.

Existe la posibilidad de entregar en fecha tardía, el jueves 15 de noviembre a las 18hs. En este caso se aplicará una penalización de 2 (dos) puntos en la nota final del trabajo y no se contará con una instancia de recuperación.

## Evaluación

Para la evaluación se tendrá en cuenta la funcionalidad y **fundamentalmente** que el diseño del conjunto las clases respete los lineamientos del paradigma de programación orientada a objetos.

No se tendrán en cuenta en la evaluación modificaciones no solicitadas que impacten en la interface visual, excepto que las mismas aporten al diseño OO.

Se debe obtener una nota mayor o igual a 4 (cuatro) para aprobar. Si se entrega un trabajo en fecha tardía que está aprobado, pero luego de aplicar la penalización pasa a estar desaprobado, el trabajo queda finalmente desaprobado.

## Consultas

Las consultas se realizarán en el horario del laboratorio, o mediante correo electrónico a la dirección **[zabaleta.laura@gmail.com](mailto:zabaleta.laura@gmail.com)**