

PROGRAMACIÓN II (CC67)

**Ciclo 2021-2**

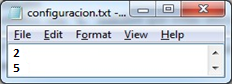
**Ejercicio Integrador**

1. ***Caminata en Marte***

## El *rover* más grande y avanzado que la NASA ha enviado a otro mundo, aterrizó en Marte el pasado jueves 18 de febrero, después de un viaje de 203 días y haber recorrido 472 millones de kilómetros. Con la llegada del rover “*Perseverance*”, se abre un camino para la llegada del hombre a Marte el 2031. Por ello se le solicita que simule la caminata de un astronauta en Marte. Durante un tiempo **“ T ”** el astronauta camina por la superficie marciana y una lluvia de meteoritos empieza después de **“ S ”** segundos. Cada vez que un meteorito colisiona con el astronauta pierde una vida. La cantidad total de vidas del astronauta es un numero aleatorio en el rango de 5 a 10.

* **La superficie de Marte**
  + Está representado por una imagen *(marte.png)*
* **El Astronauta** 
  + Es un *sprite* (*atronauta.png*)
  + Se mueve con las teclas direccionales **únicamente** por la superficie de Marte.
* **Los meteoritos**
  + Es un sprite (*meteoritos.png*) y se mueve automáticamente en forma vertical
  + Aparecen en posiciones aleatorias y en forma gradual cada “S” segundos desde la parte superior y caen en forma vertical hacia abajo
  + Si colisiona con el astronauta le quita una vida. Se contabiliza los que no colisionan y quedan en la superficie marciana.
* **“S” y “T”** son valores que deben ser leídos de un archivo plano **configuracion.txt.** El archivo es creado por el estudiante en la carpeta del proyecto.

**Ejemplo**

****

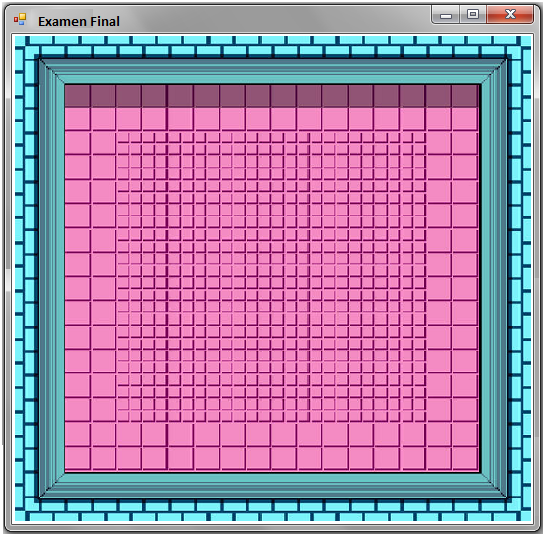
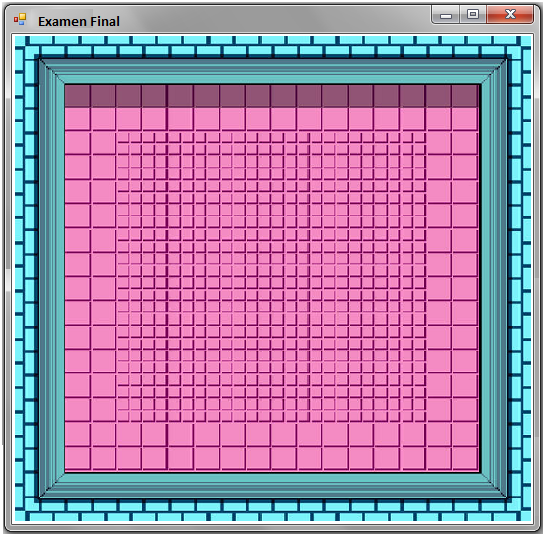


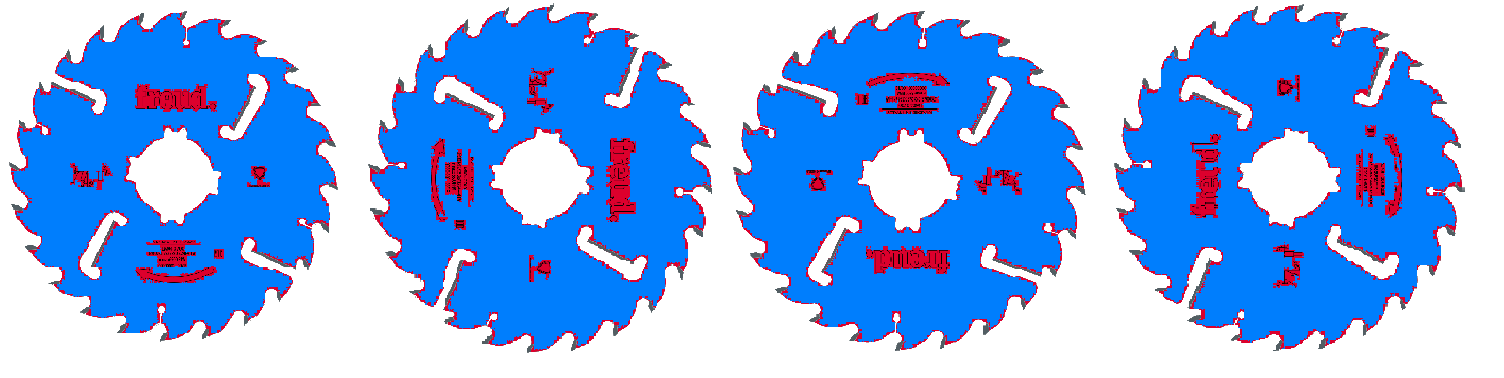
|  |  |  |
| --- | --- | --- |
|  |  |  |
| astronauta.png | marte.png | meteoritos.png |

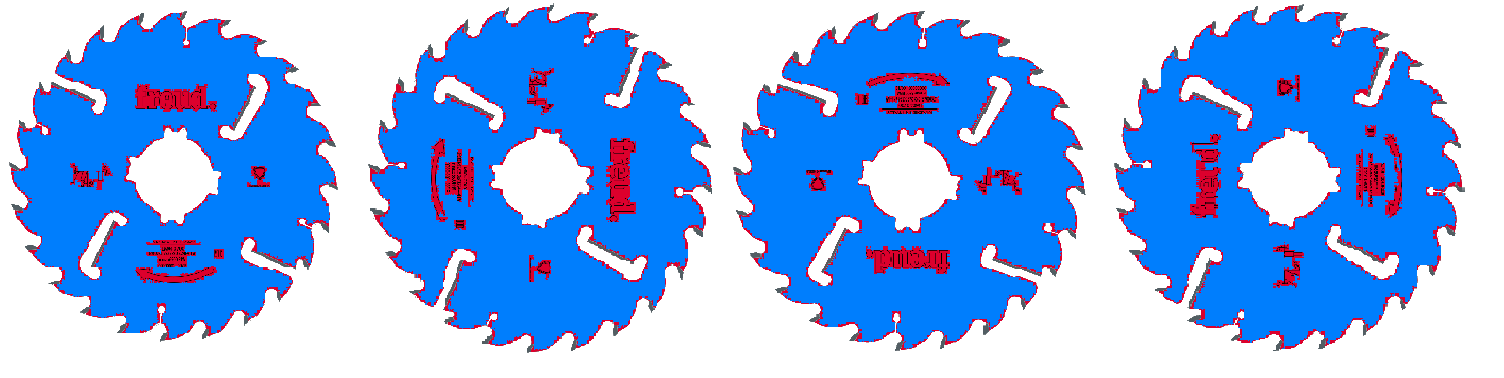
1. ***Mini Juego Disc Room***

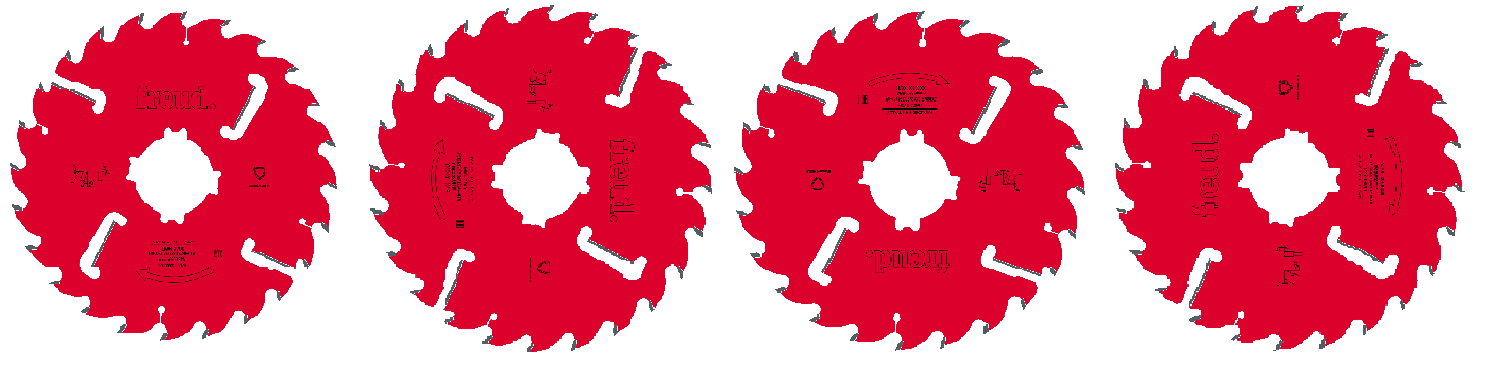
Utilizando los conceptos de POO: **Herencia, polimorfismo** y manejo de archivos, se propone realizar un programa con las funcionalidades que se describen líneas abajo y que están indicadas en la rúbrica de calificación.

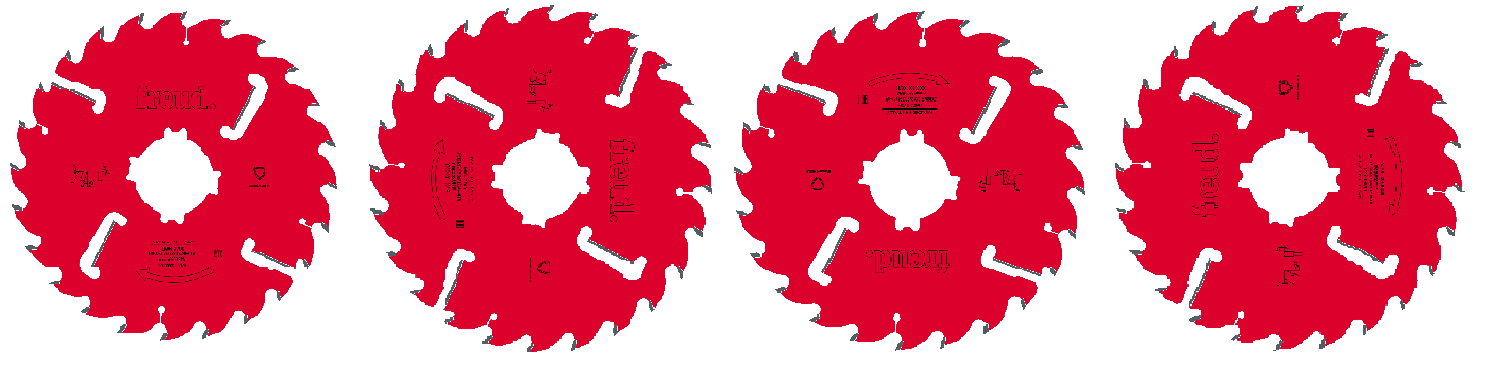
* **Diseño de Formulario**

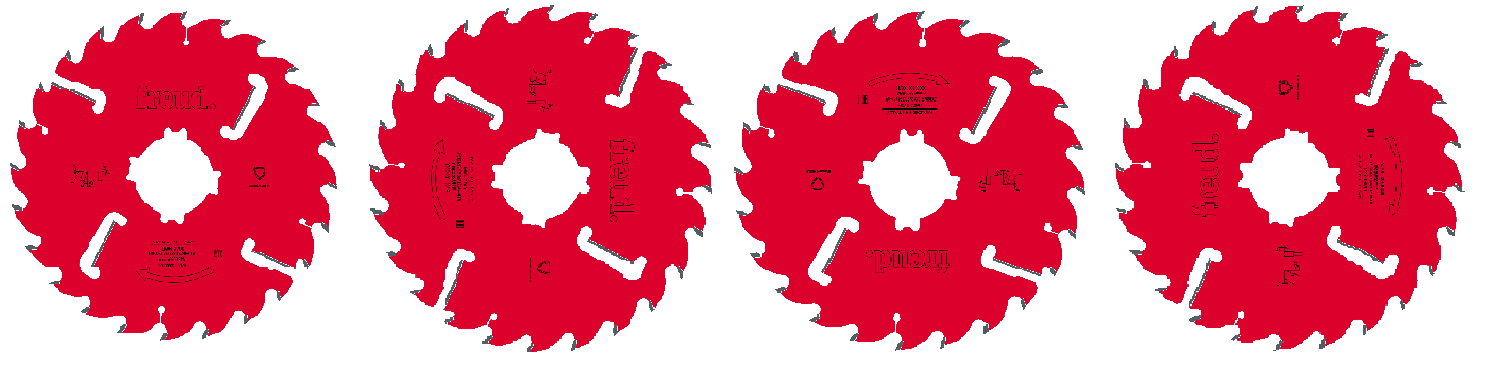
****

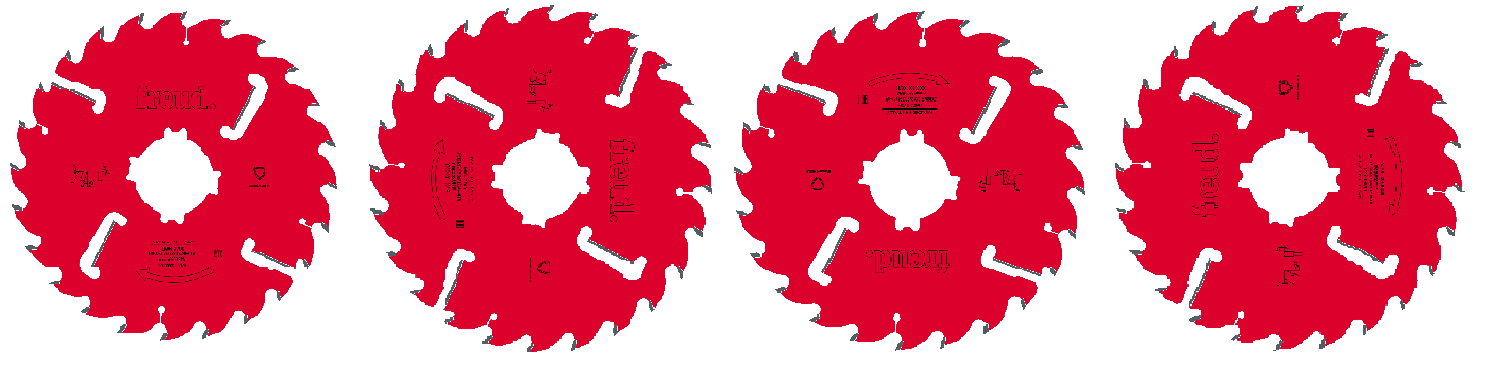
****

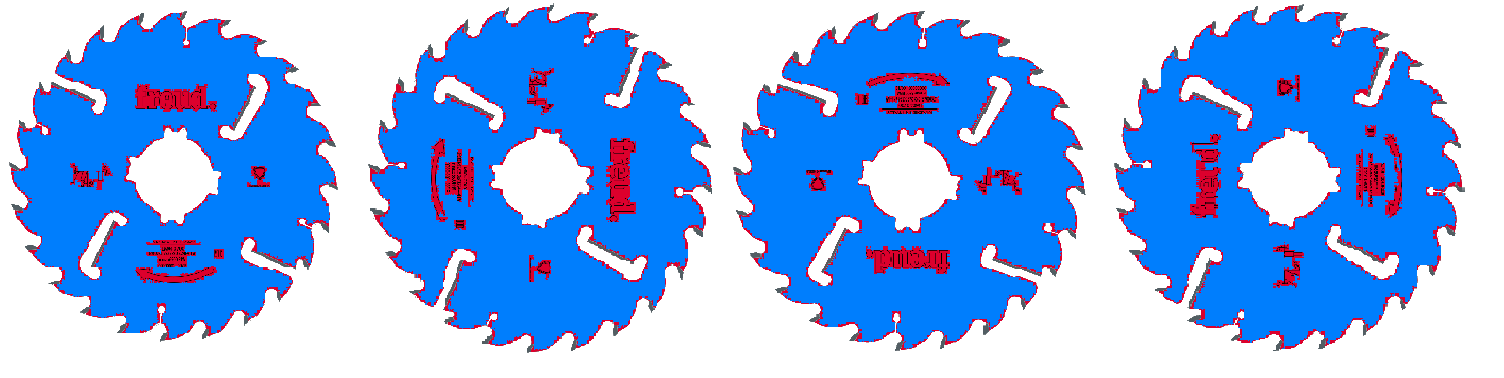






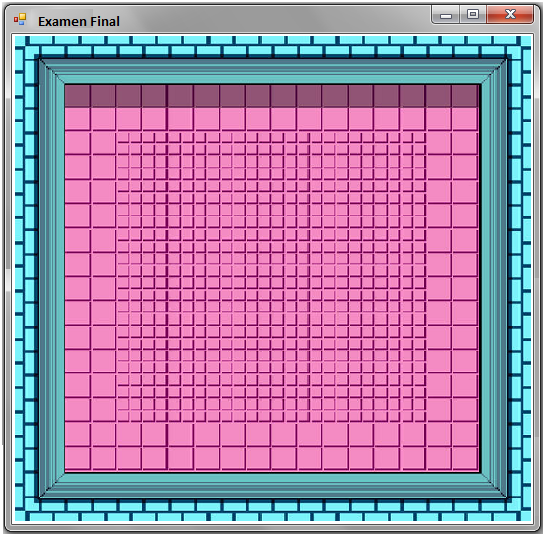


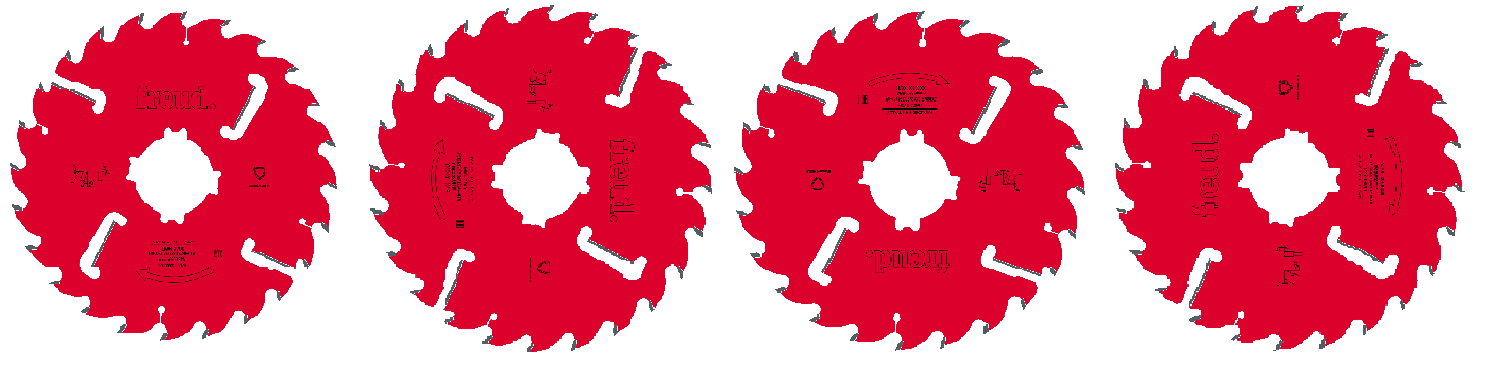




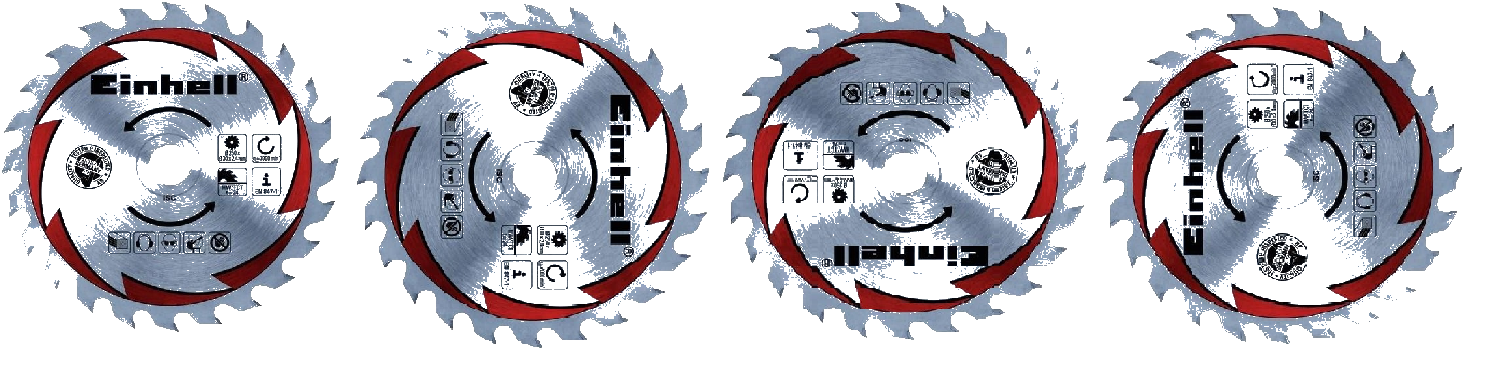
Los discos colisionan

Los discos aparecen aleatoriamente y se mueven por toda la pantalla.

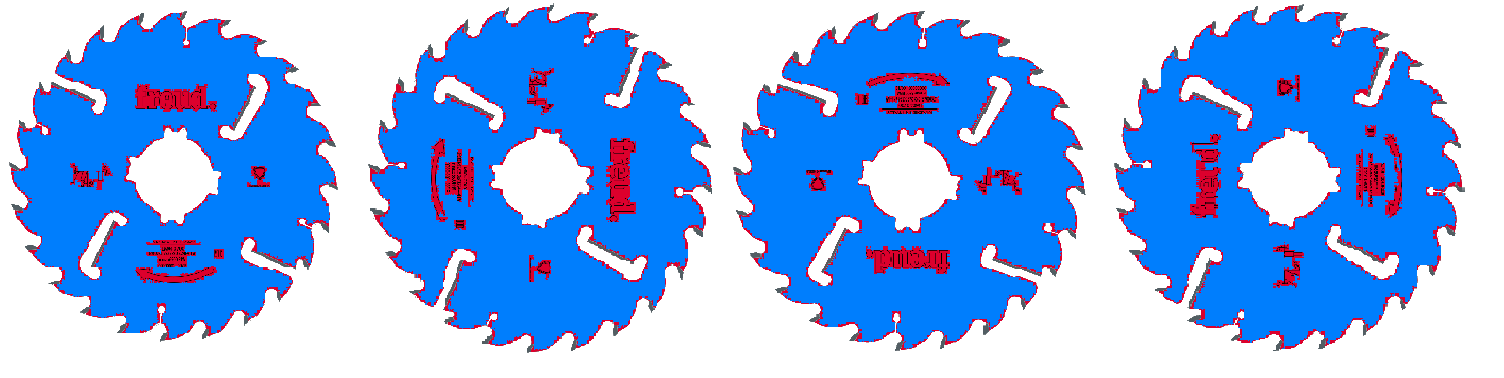
****







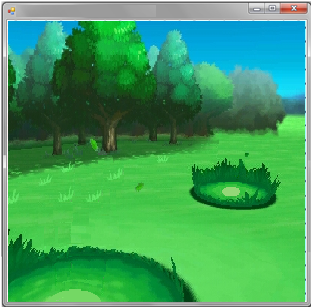
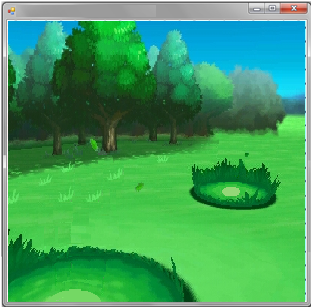
Cuando un disco azul y rojo colisionan, se genera un tercer disco

****

1. ***Mini Juego Pokemon***

Utilizando los conceptos de POO: **Herencia, polimorfismo** y manejo de archivos, se propone realizar un programa con las funcionalidades que se describen líneas abajo

* **Diseño de Formulario**

 9

**JUGADOR**



**POKEBALL**



* Se mueve aleatoriamente por toda la ventana
* **Funcionalidad**

**Jugador**

J

* + La posición inicial del jugador es “X”, “Y”.
  + El jugador se mueve con las teclas direccionales
  + Empieza con un “N” vidas.
  + El jugador debe evitar la colisión con las pokeball
    - Si el jugador colisiona con una pokeball pierde una vida y regresa a su posición inicial.
  + Si se queda sin vidas se muestra un mensaje “PERDISTE”.
  + Si el jugador logra permanecer en el escenario por “T” segundos, gana el juego y aparece un mensaje “GANASTE”

**Archivos**

Los valores “X”, “Y”, “N” y “T” son leídos de un archivo plano “*configuracion.txt*”. Si el archivo no existe debe ser creado en tiempo de ejecución.

Ejemplo

