



Universidad Nacional de La Matanza

Algoritmos y Estructuras de Datos

Trabajo Práctico Volcán en Erupción

Comisión Curso de Verano 2026

3er cuatrimestre 2025



UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

Requerimiento

Se debe diseñar e implementar un juego denominado Volcán en Erupción. La historia del juego es la siguiente:

Un antiguo volcán ha despertado.

En su interior se extiende una compleja red de cámaras conectadas por múltiples conductos. Algunas cámaras terminan en callejones sin salida; otras se ramifican en muchas direcciones impredecibles.

Criaturas acechan en las profundidades y la lava sube y baja sin previo aviso. El volcán evoluciona con el tiempo, sin esperar al explorador.

Cada turno cuenta.

*El volcán **no se detiene**.*

Reglas del juego

El jugador (**J**) debe moverse entre las cámaras del volcán persiguiendo el objetivo de poder salir por el cráter (**S**), que es la abertura principal en la cumbre, sin ser alcanzado por algunas de las criaturas (**C**) que moran en el interior del volcán. El peligro no solo reside en las criaturas, el nivel de la lava (**L**) puede subir alcanzando al jugador.

Criterios generales:

- Salida (**S**): es el punto por el cual se sale del volcán y coincide con su cumbre.
- Jugador (**J**): es el componente animado a través de los eventos producidos por el usuario que se encuentra jugando. Además:
 - o Aparece en una cámara aleatoria
 - o Puede:
 - subir a la cámara padre operando la tecla **W**
 - bajar eligiendo uno de los hijos disponibles operando las teclas **A**, **S**, **D**, **F**, etc. Este caso corresponde a una cámara con cuatro conexiones hacia abajo.
 - o Cada acción consume un turno



UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

- Premios (**P**): se encuentran distribuidos a lo largo del volcán. Cuando se captura un premio se obtiene 1 punto.
- Vidas extra (**V**): cuando se captura una Vida Extra se incrementa en uno el número de vidas disponibles para continuar jugando. En caso de haber llegado al cráter y aún posea vidas, se incrementará la puntuación con cinco puntos por cada vida extra obtenida. Las vidas extra se encuentran distribuidas a lo largo del volcán.
- Criaturas (**C**): las criaturas merodean entre las cámaras del volcán y solo pueden moverse a través de los conductos que las interconectan, es decir, las criaturas se deben mover entre cámaras (**O**) que posean conexión y su objetivo es eliminar al jugador (**J**). Aparecen de manera aleatoria dentro del volcán. En una cámara puede haber más de una criatura de manera simultánea. Cuando alcancen al jugador, tanto la criatura como el jugador mueren (si hubiera más de una criatura en la cámara morirían todas). En caso de que el jugador aún posea vidas para continuar jugando, aparecerá en el mismo lugar en donde murió. En el caso de que ya no posea más vidas la partida se dará por finalizada.
- Lava (**L**): el nivel de la lava asciende desde la base del volcán hacia el cráter (la salida **S**), e invadirá las cámaras (**O**) nivel por nivel siempre y cuando tengan conexiones desde el nivel inferior hacia el inmediato superior y así sucesivamente (se puede pensar como un fluido subiendo por una red de cañerías).

Consigna

- a) Analizar el contenido de este documento para interpretar y comprender los requerimientos.
- b) Se debe implementar el juego Volcán en Erupción en lenguaje C.
- c) El desarrollo debe ser colaborativo, para ello el equipo de trabajo debe utilizar GitHub.
- d) Configuraciones iniciales: mediante un archivo de texto denominado **config.txt** será posible establecer los siguientes parámetros:
 - Altura máxima.
 - Número máximo de conexiones hijas de las cámaras.



UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

- Número de vidas iniciales.
 - Número de vidas adicionales a distribuir aleatoriamente dentro de las cámaras.
 - Número de criaturas a distribuir aleatoriamente dentro de las cámaras.
 - Número de premios a distribuir aleatoriamente dentro de las cámaras.
- e) Al iniciar el juego, se debe generar el volcán en un archivo de texto denominado **volcan.txt** a partir de los parámetros de configuración provenientes del archivo **config.txt** descrito en el ítem anterior. La representación de los componentes del juego debe realizarse con los siguientes caracteres ASCII:

- Jugador: **J**
- Salida: **S**
- Premio: **P**
- Criatura: **C**
- Vidas extra: **V**
- Cámara vacía: **O**
- Conductos: **/ \ |**
- Cámara con lava: **O[L]**
- Cámara con premio: **O[P]**
- Cámara con criatura: **O[C]**
- Cámara con vida extra: **O[V]**
- Etc.

Si tomáramos como ejemplo del archivo de texto que describe el volcán o con los siguientes valores en los parámetros:

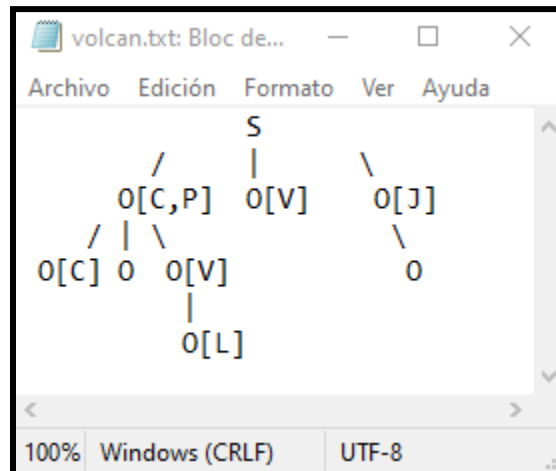
- ALTURA_MAXIMA = 4
- MAX_CONEXIONES = 3
- VIDAS_JUGADOR = 3.
- VIDAS_EXTRA = 2
- CRIATURAS = 2
- PREMIOS = 1



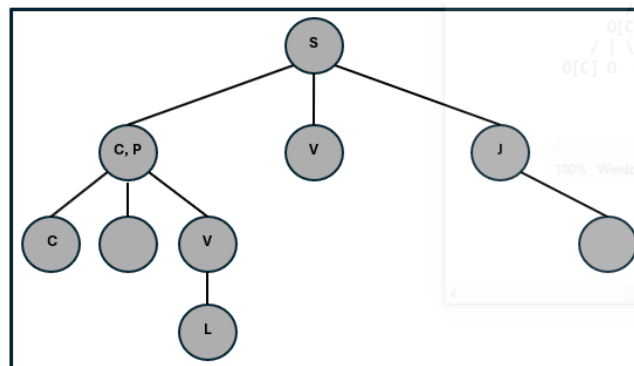
UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

Daríá como resultado un posible volcán como el siguiente:



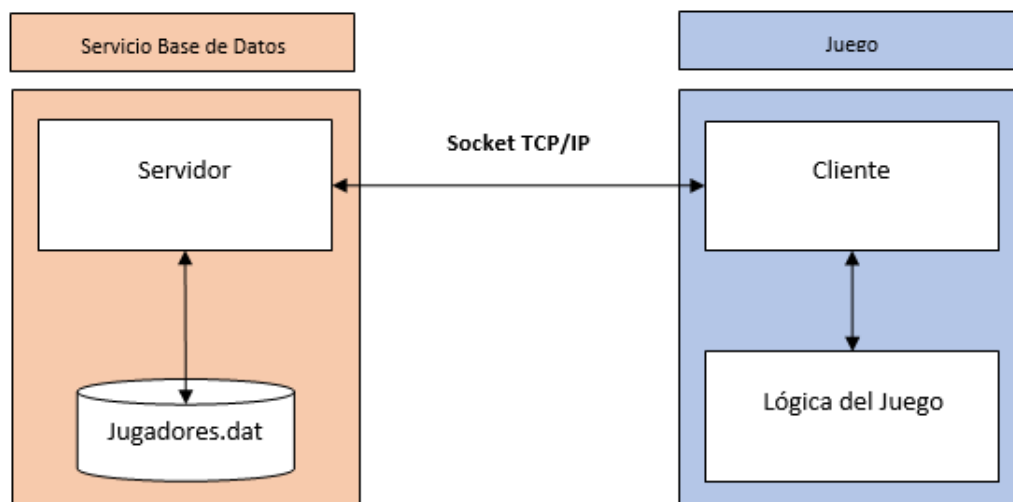
Desde el punto de vista de la representación del volcán en la computadora, tendríamos un árbol n-ario, que en este caso puntual es 3-ario (es 3-ario debido a que el parámetro MAX_CONEXIONES es igual a 3):



La raíz del árbol representa la cumbre del volcán que corresponde a la salida (**S**).

NOTA: Tener en cuenta que al iniciar una nueva partida debe crearse un **nuevo** volcán respetando los parámetros iniciales. Cada volcán que se genere debe tener una solución posible.

- f) Los movimientos del jugador y de la computadora (criaturas y nivel de lava) deben ser encolados. Luego se deben desencolar, y determinar el nuevo estado y actualizar la pantalla.
- g) Se debe llevar un registro de los movimientos realizados por el jugador. Al finalizar la partida se debe mostrar el registro de movimientos.
- h) Se debe implementar una lógica para que la computadora haga sus movimientos. Los movimientos se verán reflejados sobre las criaturas, las cuales irán tras el jugador tratando de capturarlo.
- i) El juego debe contar con un servicio de base de datos en la que se hará el registro de usuarios, partidas, puntuaciones obtenidas y cantidad de movimientos realizados en cada partida. La implementación constará de una aplicación que utilice Sockets TCP/IP. La aplicación escuchará en un puerto específico a determinar por el equipo de diseño. Las peticiones realizadas al servicio de base de datos deben ser encoladas. En la figura se muestra un esquema del requerimiento. El archivo de jugadores deberá accederse a través de un índice implementado sobre un árbol de búsqueda binaria. El índice debe mantenerse actualizado y además se debe persistir sobre un archivo binario. En caso de que el servicio se encuentre caído debe contemplarse una contingencia para jugar.



**UNLaM****Dto. Ingeniería e Investigaciones Tecnológicas**

- j) El juego deberá contener un menú que permita: ver el ranking de jugadores, jugar una nueva partida, salir del juego. Si el equipo de diseño considera alguna otra opción es libertad de este implementarla.
- k) Al iniciar el juego debe indicarse el nombre del jugador. En caso de no existir en la base de datos se procederá a su alta.
- l) El trabajo se encuentra abierto al agregado de características y/o funcionalidades que el equipo de diseño considere relevante. Estas modificaciones deben ser declaradas y documentadas. Al momento de la entrega del trabajo deben estar implementadas y en correcto funcionamiento.

Conformación del grupo de trabajo

Este trabajo práctico se realiza en equipos de hasta 4 personas, sin excepción. Los equipos deben ser informados a través de la plataforma MleL con un mensaje en el foro “Trabajo Práctico” con plazo máximo hasta el día Miércoles 04/02/2026 a las 23:59:59 hs. Debe indicarse el nombre del equipo y sus integrantes (apellido, nombres y DNI).

Entrega

La entrega del trabajo práctico se realizará de manera individual. Cada integrante del grupo deberá enviar desde la sección prácticas (portfolio) de MleL en la práctica “Trabajo Práctico” a todos los docentes, una copia del programa informando nuevamente en el mensaje los integrantes de su grupo.

El plazo máximo de entrega del trabajo práctico será el día Miércoles 25/02/2026 a las 23:59:59 hs a través de la plataforma MleL.

Los trabajos entregados después de la fecha - hora indicada no serán tenidos en cuenta, por lo que el equipo obtendrá como nota un ausente en la entrega y en la defensa, teniendo que utilizar la fecha de recuperatorio para regularizar su situación.

La entrega consta de un archivo con el siguiente formato:

TP_ALGORIMOS_2026_CV_{NOMBRE_DEL_GRUPO}.zip

**UNLaM**Dto. Ingeniería e Investigaciones Tecnológicas

No deberá incluir las carpetas *bin* y *obj*.

En la entrega sobre la plataforma MleL, se deberá adjuntar el **archivo .zip** y la **URL al repositorio GitHub**. El repositorio debe ser **público**. El contenido del repositorio debe coincidir con lo entregado en MleL.

Condiciones mínimas para lograr la aprobación de la entrega del TP (habilita a rendir la defensa)

- El producto entregado deberá cumplir con los requerimientos y las consignas indicadas en el trabajo.
- Debe compilar con 0 errores 0 warnings
- Código prolijo, dividido en funciones
- Proyecto dividido en archivos .h y .c.
- Funciones lo más genéricas posibles
- Nombres significativos de variables
- Todas las estructuras de datos creadas para la solución deberán ser un TDA, y todos los TDA deben ser genéricos. Esta condición es EXCLUYENTE.
- Deben trabajar sobre el repositorio de GitHub.

Condiciones mínimas para lograr la aprobación del TP

- aprobar la entrega del TP.
- aprobar la defensa del TP (La defensa del TP equivale a la segunda instancia de evaluación (parcial 2) de la asignatura. En caso de aprobar la entrega del TP, estará habilitado para rendir la segunda instancia de evaluación (parcial 2 - Lunes 2/3/2026). En caso de no aprobar la entrega y tener que reentregar, al aprobar en la reentrega, pasará a rendir la defensa del TP en el instancia de recuperación (Recuperatorio



UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

Miércoles 4/3/2026). **Recuerde que solo puede recuperar una única instancia de evaluación (parcial 1 o parcial 2).**