

UNIVERSIDAD DE SANTIAGO DE CHILE

FACULTAD DE INGENIERÍA

Departamento de Ingeniería Informática



PROYECTO “CHORIPAN ROYALE”

Camila Castillo Cárdenas, Álvaro Troncoso Soza

Fecha:

16 de octubre de 2022

Profesores:

Alejandro Cisterna

Gary Simken Ruminot

ÍNDICE

1. INTRODUCCIÓN	2
¿Qué es el lenguaje de programación C?	2
2. DESCRIPCIÓN DEL PROBLEMA	3
Figura 1: Representación visual del mapa del videojuego.	3
Figura 2: Representación conceptual del mapa.	4
Figura 3: Representación conceptual de un personaje con sus respectivas zonas de daño.	5
3. DESCRIPCIÓN DE LA SOLUCIÓN	7
Figura 4: Archivo de las armas (Ejemplo)	7
Figura 5: Archivo del mapa (Ejemplo)	8
4. CONCLUSIONES	9
5. REFERENCIAS	10
6. ANEXO	10
Anexo 1: Organización del tiempo de proyecto:	10

1. INTRODUCCIÓN

En este informe se presenta el enunciado del proyecto del laboratorio a construir por estudiantes del curso de Métodos de Programación para el segundo semestre del año 2022.

Este proyecto tiene fines educacionales, de tal manera que los estudiantes busquen distintas técnicas de resolución de problemas y, por parte del cuerpo académico, se espera que los estudiantes puedan plantear soluciones con todos los conocimientos que se van adquiriendo en las clases de teoría, laboratorio y por los ejercicios que se van haciendo en dichas clases, en conjunto con el ingenio e imaginación de los estudiantes.

¿Qué es el lenguaje de programación C?

El lenguaje de programación C es apreciado por la eficiencia del código que produce y es el lenguaje de programación más popular para crear software de sistemas, aunque también se utiliza para crear aplicaciones hoy en día. (Buriticá, 2017)

Por otra parte, un paradigma de programación es un modelo que proporciona los elementos matemáticos y técnicos para poder aprovechar un lenguaje de programación en la solución de un problema. (Buriticá, 2017). Entonces, un paradigma de programación proveerá la visión y métodos que un programador utilizará en la construcción del programa.

Entendiendo lo que es un paradigma de programación, se puede comprender que existen diferentes paradigmas en la programación. Uno de ellos, es el paradigma imperativo procedural, el cual divide al programa como un conjunto de bloques de código ejecutable, llamados "procedimientos". Esta forma de abstracción logra que el programador pueda utilizar la aproximación "divide y conquistarás", para diseñar un flujo de control de los programas y que la solución sea más abordable. (Pérez & López, 2007)

El objetivo principal de este informe es caracterizar y describir detalladamente el videojuego a realizar, de acuerdo con cada una de sus funcionalidades. El videojuego por realizar se implementará en el tipo de lenguaje C, con un paradigma de programación imperativo procedural.

Este informe tendrá la siguiente estructura: Primero, se describirán exhaustivamente las características del videojuego a realizar en términos de mapa, los jugadores, las armas y utilidades y por último, la mecánica del juego. Segundo, se describirán, en términos prácticos, cómo se llevará a cabo la implementación del videojuego, las entradas, las salidas y cómo funcionará la aplicación a grandes rasgos. Tercero, se exponen las conclusiones del trabajo realizado. Por último, se adjuntan los respectivos Anexos.

2. DESCRIPCIÓN DEL PROBLEMA

Este proyecto tiene como fin el desarrollo de un videojuego basado en lenguaje C, para una competición interna de los cursos de métodos de programación de la Universidad de Santiago de Chile. La idea es que en grupos de 2 estudiantes, se realice un videojuego estilo battle royale, el cual tomará como nombre "Choripan Royale". En la Figura 1, una representación visual del mapa del videojuego.



Figura 1: Representación visual del mapa del videojuego.

Se realizarán dos versiones de este videojuego, a modo de que uno se pueda jugar de manera local y otro que pueda jugar de modo automático. Esta última versión del juego tendrá el fin de poderse jugar en un mapa entregado por los profesores y donde se enfrenten a las demás secciones del curso.

A continuación, se explicarán de manera detallada las características del mapa, las armas y utilidades, y la mecánica del juego.

a) Mapa:

El mapa del videojuego consiste en una simulación de una ciudad en guerra. Este mapa está separado por bloques de 1x1. En cada uno de estos bloques se puede encontrar un jugador, o un arma, o una utilidad o un obstáculo. En la figura 2, se muestra una representación conceptual del mapa de "Choripan Royale".

X	1	#	#	#	#
A	P	2	5	4	B
3	3	2	X	3	1
4	#	4	1	1	#
3	4	5	Y	2	1
X	X	X	1	J	1
4	T	2	#	2	1
5	5	2	1	3	X
#	#	#	1	3	2
K	4	5	C	2	3

Figura 2: Representación conceptual del mapa.

Se debe tener en cuenta que los jugadores no podrán estar en un bloque por mucho tiempo, ya que los bloques tendrán contadores para saber cuántos jugadores han pasado por ahí. Entonces, si un jugador no se mueve, el contador disminuye 1 al pasar el turno. Si el contador llega a 0, el bloque será bombardeado e inhabilitado. Si durante este bombardeo existe un jugador en aquel bloque, el jugador quedará eliminado.

Como ya se mencionó, la representación conceptual del mapa será por bloques de 1x1. Cada bloque tendrá una letra, símbolo o número, los cuales se explicarán a continuación:

- X: Bloque bombardeado
- #: Obstáculo para los jugadores
- A, B, C: Jugadores
- J, K, P, T, Y, Z: Armas
- 1,2,3,4,5: Contador de bloque.

b) Jugador:

Cada personaje es representado por 9 bloques. Cada uno de estos bloques tiene un número del 1 al 3. El número 1 representa al área más alta en daño, el número 2 representa el daño normal y el número 3 representa un daño bajo. El daño recibido en cada una de estas zonas dependerá también del tipo de arma que se utilice. En la figura 3, se muestra una representación conceptual de lo que sería un jugador como bloque de 9x9 y sus respectivas zonas de daño.

3	2	3
2	1	2
3	2	3

Figura 3: Representación conceptual de un personaje con sus respectivas zonas de daño.

c) Armas y utilidades:

A continuación se detallan cada una de las armas presentes en el juego:

1) Chori-pipa (J):

- Ventajas: Daño alto, disparo instantáneo, alcance de 10 bloques.
- Desventajas: Demora en recarga.

2) Chori-digul(K):

- Ventajas: Daño normal, alcance de 6 bloques (División entera).
- Desventajas: Disparo normal, baja munición.

3) Chori-plomo(P):

- Ventajas: Daño alto, disparo instantáneo.
- Desventajas: Demora en recarga, alcance de 4 bloques (División entera).

4) Electro-chorizo(T):

- Ventajas: Daño alto, disparo instantáneo.
- Desventajas: Alcance de 1 bloque.

5) Chorizo táctico (C):

- Ventajas: Daño alto, ataque instantáneo.
- Desventajas: Alcance de 1 bloque, sólo se puede utilizar cuando se está frente a frente con otro jugador.

Cabe destacar que cada jugador tiene un arma por defecto, la cual es la Chorizo-táctico, la cual es un arma cuerpo a cuerpo, ya que tiene alcance de 1 bloque y solo se utiliza cuando se está frente a otro jugador. En el mapa, pueden estar cualquiera de los otros tipos de armas.

Cada una de las armas, puede estar o no estar en la partida y puede estar en más de un bloque del mapa a la vez.

Los tipos de disparo, daño y recarga se explican detalladamente a continuación:

- Disparo normal: La trayectoria de la bala avanza bloque por bloque en cada turno.
- Disparo instantáneo: La trayectoria de la bala llega directo a su objetivo. No avanza bloque por bloque.
- Tiempo de recarga: Se considera en 4 turnos. El jugador dispara, el enemigo realiza su jugada, el jugador realiza su jugada, el enemigo realiza su jugada y el arma se puede volver a utilizar. No se imposibilita la habilidad del jugador de desplazarse
- Daño alto: Puede inhabilitar al jugador si toca la zona 1 o 2 del jugador, Si toca la zona 3, inflige una cantidad considerable de daño.
- Daño normal: No inhabilita a un jugador con un disparo si toca en la zona 2 del jugador y tampoco es capaz de hacer daño en la zona 1 del jugador. Si el disparo toca la zona 3 del contrincante, inflige una cantidad mínima de daño.
- Ningún arma puede atravesar obstáculos, es decir, que para poder hacer daño, se debe encontrar en la línea de visión del jugador. Si un jugador es inhabilitado, no podrá continuar en la partida.

En el mapa también se podrán encontrar 2 objetos de utilidad, los cuales son:

- Chorzepan (Y): Devuelve un 50% de la salud del personaje
- Choriarmor (Z): Restablece el 100% del blindaje del personaje.

d) Mecánica del juego.

En términos de jugabilidad, el battle royale tendrá las siguientes 10 reglas:

1. Los jugadores solo comienzan con el arma cuerpo a cuerpo.
2. Cada jugador solo puede llevar 1 arma extra ("CHORIZO TÁCTICO" + otra).
3. No es obligatorio recoger un arma.
4. Para ganar se debe inhabilitar a los demás jugadores.
5. El orden de movimiento inicial debe ser decidido por un dado.
6. Deben existir al menos 3 y como máximo 9 armas dentro del terreno de juego.
7. Las armas deben estar distribuidas de manera equitativa, es decir, que se debe encontrar al menos 1 arma en un radio de 3 bloques de distancia con cada jugador.
8. Cada jugador puede llevar un máximo 2 utilidades de las que se encuentran en el mapa.
9. No es necesario recoger los objetos especiales del mapa.
10. Las balas no pueden atravesar los obstáculos

3. DESCRIPCIÓN DE LA SOLUCIÓN

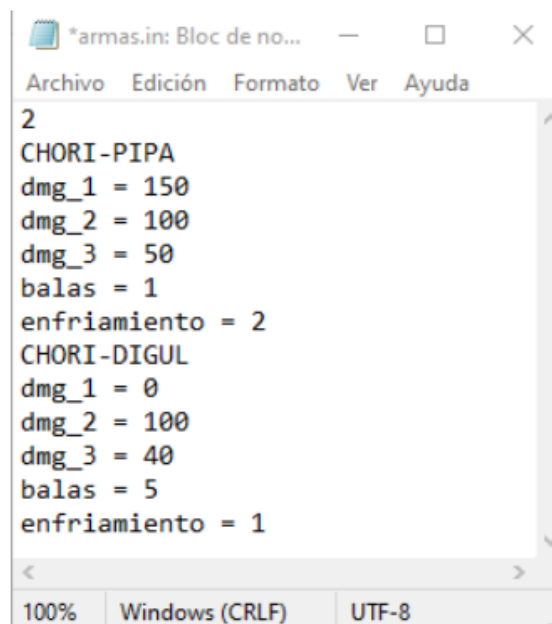
El proyecto está basado en la resolución de una situación ficticia, la cual debe ser programada en lenguaje C, contemplando que la interacción con el usuario debe ser mediante los dispositivos de entrada y salida estándar del computador y archivos de texto plano. También se aceptan otro tipo de visualización más sofisticada, si es que así se desea por el equipo desarrollador.

Para las entradas del programa se realizarán 2 archivos:

- Las armas
- El mapa

Para las armas, se le presentará un archivo llamado “armas.in”, en el cual se leerán los datos de cada una de las armas, como lo son su nombre, y sus características (daño en las esquinas, al centro, balas, tiempo de recarga, tipo de disparo). El archivo estará compuesto por N líneas que se agrupan de a 6, dónde la primera línea del archivo indica la cantidad de tipos de armas que se deben leer, para que luego las siguientes 6 tengan el siguiente formato:

1. Nombre del tipo de arma.
2. Daño que inflige al impactar al jugador según zona 1, 2 ó 3.
3. Balas, Cuántas balas puede disparar sin recargar.
4. Cuantos turnos el jugador queda inhabilitado para disparar mientras recarga.
5. Tipo de disparo: normal o instantáneo.



```
*armas.in: Bloc de no...
Archivo  Edición  Formato  Ver  Ayuda
2
CHORI-PIPA
dmg_1 = 150
dmg_2 = 100
dmg_3 = 50
balas = 1
enfriamiento = 2
CHORI-DIGUL
dmg_1 = 0
dmg_2 = 100
dmg_3 = 40
balas = 5
enfriamiento = 1
100%  Windows (CRLF)  UTF-8
```

Figura 4: Archivo de las armas (Ejemplo)

Como se mencionó previamente, el mapa del juego tiene ciertas letras y números que representan un elemento en particular. Sin embargo, como cada partida debe ser distinta, no existe un orden preestablecido de cómo será la distribución del mapa. Por ende, para la creación del mapa se tendrá el archivo llamado “mapa.in”, en el cual se tendrá que extraer la siguiente información:

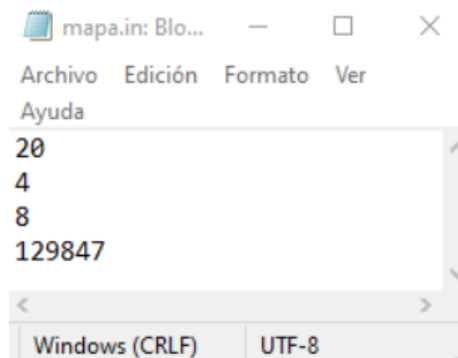


Figura 5: Archivo del mapa (Ejemplo)

- Dimensiones del mapa
- Cantidad de jugadores
- Cantidad de obstáculos
- Semilla para obtención de valores aleatorios.

Cada uno de los elementos antes mencionados, se encontrará en una línea distinta del archivo. Además, se encuentra un valor en particular el cual es la semilla 1, que tendrán que utilizar para establecer los parámetros principales del juego (cantidad de armas presentes en la partida, límite de turnos en cada bloque y posición inicial de cada objeto y jugador).

Cabe mencionar que el mapa es de tamaño NxN, donde “N” se obtiene a partir del archivo y además la variable “N” debe tener como valor mínimo 20. Además, al momento de posicionar a cada jugador en el mapa, se debe verificar que este no quede encerrado por obstáculos, es otras palabras, que tenga al menos la posibilidad de realizar un movimiento.

En cuanto a las salidas, dado que el presente juego es del estilo competitivo, puede que varios jugadores crean que existen injusticias a la hora de estar interactuando, es por esto que se pide un registro de acciones que se hayan realizado durante la partida. La salida será un archivo de texto que debe incluir en cada fila del archivo uno de los pasos realizados bajo el formato “posición anterior, posición actual, acción a realizar”. En la última fila del archivo, se debe indicar qué jugador fue el ganador de la partida.

Las posibles acciones por realizar son las siguientes:

- Movimiento del jugador.
- Obtener las armas.
- Disparar

- Recargar
- Recibir daño (JX: -HP)
- Fin del Juego o para hacerlo más dinámico "Game Over"

Implementación práctica:

Para resolver la problemática, se creará una aplicación a partir de un código basado en lenguaje C y en las características del videojuego anteriormente nombradas.

La aplicación antes mencionada que se realizará en este proyecto tendrá por nombre "*Choripan Royale: Sangre, grasa y honor*". Se añadirán dos nuevas armas y se describirán con todas sus características. Además, se generarán objetos coherentes con la simulación y serán utilizados por los jugadores una cantidad "n" de veces. La aplicación mostrará las estadísticas del jugador, su vida, blindaje, armas y el daño que inflige cada arma.

La aplicación podrá ser una simulación automática del juego, donde el usuario decidirá si quiere manejar al jugador de manera manual. Se permitirá al usuario poder utilizar objetos de un solo uso y poder tener equipados objetos de más de un uso. También se podrá ver la cantidad de usos según munición que posee el arma y permitirá al usuario ver todas las cosas dentro del mapa.

La aplicación también permitirá al usuario terminar la partida y entrar a otra, si corresponde. El usuario podrá salir del juego en cualquier momento en que lo desee. Se tendrá un log de acciones, en el cual se indicará qué acción se realizó, el tiempo en el que se realizó, si fue el usuario o la aplicación quién realizó la acción. El Log de acciones se escribirá en un archivo de texto, donde se podrá ver el año, el mes, el día, la hora y los minutos en el que se realiza la aplicación.

Además, los datos del personaje se guardarán en un struct. Se guardará la ID, la posición en X e Y, la vida, el blindaje, las armas y objetos (como arreglos), la dirección (arriba=1, abajo=2, derecha=3, izquierda=4) y por último, la acción realizada (movimiento =1, disparar arma=2, tomar/cambiar arma=3, utilizar objeto=4, tomar/cambiar objeto=5, ninguna acción=0)

4. CONCLUSIONES

A lo largo de este informe se ha podido observar las bases de lo que será el videojuego, con sus debidas características. La idea es poder generar una aplicación en el lenguaje de programación C considerando las diversas funcionalidades correspondientes. En esta entrega, se ha logrado cumplir el objetivo de definir la aplicación y sus diversas funcionalidades, en conjunto con el nombre de la aplicación. Además, se ha podido observar a grandes rasgos la implementación práctica de la aplicación.

Aun así, se puede considerar que esto es solo el comienzo, y que queda aún generar todo lo descrito anteriormente en la sección implementación práctica, donde se debe crear la aplicación en sí. Se espera que en la segunda entrega se pueda avanzar lo más posible en lo que concierne a lo conceptual de la aplicación, de tal manera que cuando llegue el momento de generar el código, todo lo conceptual esté resuelto y comprendido.

5.REFERENCIAS

Buriticá, O. I. T. (2017). *Programación imperativa con lenguaje C*. Ecoe Ediciones.

Pérez, Y. P., & López, L. M. (2007). Multiparadigma en la enseñanza de la programación. In *IX Workshop de Investigadores en Ciencias de la Computación*.

6. ANEXO

Anexo 1: Organización del tiempo de proyecto:

A continuación, se muestran las respectivas fechas de entrega a tener en consideración para la organización del tiempo:

- Fecha de Entrega 1 - 16/10/2022 a las 23:59.
 - *Se utilizó un tiempo de aproximadamente 4 horas, por cada integrante del equipo.*
- Fecha de Entrega 2 - 30/10/2022 a las 23:59.
 - *Quedan 14 días para la entrega 2. Se esperan reuniones de 1 hora 2 veces por semana.*
- Fecha de Entrega 3 - 29/11/2022 a las 23:59
 - *Se organizará el tiempo en durante la entrega 2, acorde a la carga académica de cada integrante.*
- Fecha de Entrega 4 - 13/12/2022 a las 23:59
 - *Se organizará el tiempo durante la entrega 3, acorde a la carga académica de cada integrante.*