# Trabajo práctico 3: Implementación (v1.0)

# ExtremeExorcism

# Normativa

Límite de entrega: La entrega se coordina con su corrector/a del Trabajo Práctico 2. La misma debe ser primero digital para luego coordinar una revisión presencial.

Normas de entrega: Ver "Información sobre la cursada" en el sitio Web de la materia.

(http://campus.exactas.uba.ar)

# 1. Enunciado

Este TP consiste en implementar en C++ todos los módulos correspondientes al juego diseñado en el TP 2. El código debería respetar el diseño elegido en el TP 2 de la manera más fiel posible. Obviamente se permite y se espera que corrijan todos los potenciales bugs que puedan llegar a encontrar en el diseño así como adaptaciones requeridas durante la corrección del TP 2.

#### 1.1. Módulos básicos

Pueden utilizar las siguientes clases de la STL para los respectivos módulos básicos:

| Módulo         | Clase       |
|----------------|-------------|
| Lista Enlazada | std::list   |
| Pila           | std::stack  |
| Cola           | std::queue  |
| Vector         | std::vector |

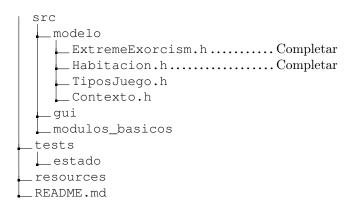
Además se proveen implenentaciones de la cátedra de Conjunto y Diccionario lineal bajo el namespace algo2:

| Módulo             | Clase             |
|--------------------|-------------------|
| Diccionario Lineal | algo2::linear_map |
| Conjunto Lineal    | algo2::linear_set |

## 1.2. Detalles de implementación

Como es usual en los trabajos del laboratorio, para resolver el trabajo práctico se les provee un proyecto de C++ a completar. El mismo puede ser utilizado desde CLion usando el archivo CMakeLists.txt como descripción del proyecto.

El proyecto tiene la siguiente estructura de directorios:



En el directorio src/gui se encuentran los archivos que hacen al funcionamiento de la interfaz gráfica. En src/modulos\_basicos se encuentran los módulos provistos por la cátedra. .cpp y .hpp. En src/modelo se encuentran las interfaces para el modelo del juego. En esta carpeta se encuentran los archivos

ExtremeExorcism.h y Habitacion.h que proveen una interfaz común para estos dos módulos para ser usados por los tests y la interfaz gráfica. Además de implementar los módulos diseñados en el TP2, deberán completar las clases definidas en estos dos archivos para que utilicen sus módulos y provean la interfaz dada. En src/TiposJuego.h se presentan implementaciones de los tipos básicos del juego. Al adaptar sus módulos a las interfaces comunes de class ExtremeExorcism y class Habitacion, deberán usar estos tipos. Si así lo desean, pueden utilizar estos mismos para la implementación de sus módulos. Contexto.h provee la class Contexto que posee la operación localizar\_jugadores. Esta misma tienen una interfaz distinta pero más sencilla que la dada en el TP2.

resources posee los archivos con las imágenes para la interfaz gráfica así como un archivo m1.txt con la descripción del mapa.

A razón de la variedad de interfaces posibles a partir del enunciado del tp2, para las operaciones que permiten observar el estado de los Fantasmas y Jugadores, estas se unificaron de la siguiente manera:

```
list<pair<Jugador, PosYDir>> posicionJugadores() const;
list<PosYDir> posicionFantasmas() const;
PosYDir posicionEspecial() const;
list<PosYDir> disparosFantasmas() const;
```

Estas permiten construir las listas resultados a partir de las interfaces de sus módulos.

# 1.3. Testing

El trabajo práctico cuenta con tests de la cátedra que su implementación deberá pasar. Estos mismos se encuentran en la carpeta tests. Los tests se pueden correr desde CLion con el objetivo correrTests o desde consola siguiendo los pasos explicados en README.md.

# 1.4. Interfaz gráfica

Junto con el código provisto se presenta código auxiliar para ejecutar el juego con interfaz gráfica. En README.md se encuentran las instrucciones para instalar dependencias, compilar y correr. También se puede ejecutar en CLion con el objetivo ExtremeExorcism. La interfaz gráfica solo fue probada en Ubuntu.