

# Desarrollo del juego Ajedrez en C++

Benanny Cjuno  
Escuela Profesional de Ingeniería de  
Sistemas  
Universidad Nacional de San Agustín  
Arequipa Peru  
bcjunom@unsa.edu.pe

Oscar Cuadros  
Instituto de Ciencias matemáticas de  
Computación  
Sao Paulo Brasil  
ocuadros1@unsa.edu.pe

**Resumen**—El ajedrez es un juego milenario de estrategia, donde participan dos jugadores, cada uno de ellos controla un equipo de fichas, cada una con características distintas, donde el objetivo principal es capturar la ficha más importante de cada jugador, el Rey, al momento de que el Rey es capturado el juego termina. También tenemos la programación orientada a objetos que busca tener una mejor calidad de software con eficiencia y eficacia, viendo del punto de vista de la programación podemos tomar cada una de las fichas como un objeto y con atributos y características similares y diferentes. En el presente trabajo desarrollamos un juego en el lenguaje de programación C++ un videojuego de ajedrez, tomando en sí las características principales de este juego, movimiento de las fichas y funciones que tienen en el tablero de ajedrez, podemos concluir que la mejor técnica para desarrollar este juego es la programación orientada a objetos, donde se desarrollan múltiples funcionalidades.

**Palabras clave**—C++, objeto, Ajedrez.

**Abstrac**— Chess is a millennial game of strategy, where two players participate, each of them controls a team of tiles, each with different characteristics, where the main objective is to capture the most important tile of each player, the King, at the time of As the King is captured the game ends. We also have object-oriented programming that seeks to have a better quality of software with efficiency and effectiveness, seeing the programming point of view we can take each of the cards as an object and with similar and different attributes and characteristics. In the present work we develop a game in the C++ programming language, a chess videogame, taking in itself the main characteristics of this game, movement of the pieces and functions that they have on the chessboard, we can conclude that the best technique To develop this game is object-oriented programming, where multiple functionalities are developed.

**Keywords**—C++, object, Chess.

## I. INTRODUCCION

### A. Programacion Orientada a Objetos

Podemos definir que a la programación orientada a objetos (POO) como una técnica moderna para disminuir el coste del software, mejorar la eficiencia en la programación y reducir el tiempo que se necesita para desarrollar una aplicación. Es un modelo de programación que utiliza objetos para la solución de problemas. De esta manera se obtienen menos líneas de código y obtenemos módulos más comprensibles donde existe una relación entre cada concepto que se desarrolla y cada objeto en el desarrollo. Un programa orientado a objetos esta enfocado netamente en estos, los entendemos como entidades que tienen atributos, datos y métodos o procedimientos. Si bien podemos pensar que tiene muchas ventajas, puede

presentar inconvenientes, la ejecución de un programa no gana velocidad y obliga al usuario a aprender una biblioteca de clases antes de comenzar a usar un lenguaje orientado a objetos [1].

### B. Ajedrez

Es un juego entre dos contrincantes en el que cada uno dispone de 16 piezas móviles que se colocan sobre un tablero, dividido en 64 casillas o escaques. En su versión de competición, está considerado como un deporte4 aunque en la actualidad tiene claramente una dimensión social, educativa y terapéutica[2],

Se juega sobre un tablero cuadrículado de 8×8 casillas (llamadas escaques), alternadas en colores blanco y negro, que constituyen las 64 posibles posiciones de las piezas para el desarrollo del juego. Al principio del juego cada jugador tiene dieciséis piezas: un rey, una dama, dos alfiles, dos caballos, dos torres y ocho peones. Se trata de un juego de estrategia en el que el objetivo es «derrocar» al rey del oponente. Esto se hace amenazando la casilla que ocupa el rey con alguna de las piezas propias sin que el otro jugador pueda proteger a su rey interponiendo una pieza entre su rey y la pieza que lo amenaza, mover su rey a un escaque libre o capturar a la pieza que lo está amenazando, lo que trae como resultado es el jaque mate y el fin de la partida.[2]



Fig.1 Tablero de Ajedrez

Fuente wikipedia

## II. HERRAMIENTAS

### A. QTCreator

Es un IDE multiplataforma programado en C++, JavaScript y QML creado por Trolltech el cual es parte de SDK para el desarrollo de aplicaciones con Interfaces Gráficas de Usuario (GUI por sus siglas en inglés) con las bibliotecas Qt, Los sistemas operativos que soporta en forma oficial son: [3]

- GNU/Linux 2.6.x, para versiones de 32 y 64 bits con Qt 4.x instalado. Además hay una versión para Linux con gcc 3.3

- MAC OS X 10.4 o superior, requiriendo Qt 4.x
- Windows XP y superiores, requiriendo el compilador MinGW y Qt 4.4.3 para MinGW.



Fig.2 logo

#### B. C++

- Es un C mejorado desarrollado por Bjarne Stroustrup en los laboratorios Bell proporcionando un conjunto de características que “pulen” al lenguaje C, sin embargo lo más importante es que proporciona una capacidad orientada a objetos[4].



Fig.3 Logo

### III. METODOLOGIAS.

#### A. Programacion Orientada a Objetos

1) *Herencia*: La herencia es uno de los conceptos básicos de la programación orientada a objetos, ya que puede reutilizar variables y funciones definidas en otras clases. Para hablar de herencia, se deben introducir los conceptos de clases base y derivadas. La clase que pretende heredar sus propiedades (variables) y funciones (métodos) a otras clases se llama clase base. Por otro lado, las clases derivadas se denominan clases que se implementan reutilizando atributos y funciones heredadas de una (o más) clases base. La herencia ayuda a mejorar la escalabilidad de la aplicación hasta cierto punto, porque cuando las variables o métodos heredados deben ser modificados o eliminados en todas las clases derivadas, no es necesario tratarlos por separado en cada clase, sino directamente en la aplicación. Completado en la clase del programa. Las clases base y las clases derivadas solo heredan y se actualizan de estos miembros[4].

En C ++, la herencia se expresa en la implementación de una clase mediante el operador de dos puntos, seguido del tipo de herencia (lo presentaremos más adelante) y el nombre

de la clase base de la que se heredarán. Vale la pena señalar que el tipo de herencia predeterminado en C ++ es herencia privada sin especificar el campo explícitamente. Esta es la implementación más simple del concepto de herencia en C++[4]:

2) *Polimorfismo*: Esta técnica de programación orientada a objetos nos ha permitido trabajar con mayor facilidad al momento de trabajar con las distintas clases que tenemos, podemos escribir los objetos en forma genérica, como objetos de clase base para un conjunto de clases existentes, mediante una jerarquía, en este caso tenemos una clase general la clase Pieza, y las demás clases que están bajo mediante jerarquía que son cada una de las piezas del ajedrez, en este caso: Peon, Caballo, Alfil, Torre, Dama y Rey.

Como sabemos entonces las distintas piezas son derivadas de la clase Pieza, como sabemos todas tienen características similares en este caso un método donde se agregue la imagen a cada una de las piezas y también agregar los movimientos, en este caso todos cuentan con distintas imágenes y movimientos, pero en general realizan las mismas acciones. Al momento de llamar a la función de setImagen que pertenece a la clase base Pieza y de esa forma el programa determina en tiempo de ejecución que función de setImagen de la clase derivada se va a utilizar. Este comportamiento se consigue cuando declaramos en la clase base la función como virtual y en cada una de las clases derivadas se van sobreponiendo para colocar en este caso la imagen adecuada o en caso contrario también se van a colocar los movimientos que pueden realizar cada una de las piezas del juego de ajedrez.[4].

#### IV. Uso

Tenemos la primera interfaz donde podemos seleccionar el modo en el cual deseamos jugar, para los movimientos de las diversas piezas estas ya están programadas, entonces al momento de seleccionar una pieza nos mostrará los posibles movimientos, unos ejemplos de esta funcionalidad se encuentra en las figuras 16 y 17 y finalmente tenemos el jaque, cuando el rey del oponente está amenazado por una de nuestras piezas entonces se pintará la casilla de color azul,

#### REFERENCES

- [1] Programacion Orientado a Objetos. [https://es.wikipedia.org/wiki/Programación\\_orientada\\_a\\_objetos](https://es.wikipedia.org/wiki/Programación_orientada_a_objetos)
- [2] Programacion Orientado a Objetos. <https://es.wikipedia.org/wiki/Ajedrez>
- [3] Ezust, A., & Ezust, P. (2012). An introduction to design patterns in C++ with Qt (Vol. 1). Prentice Hall.
- [4] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.