

**Informe sobre contenidos
comunicativos
GA4-240201524-AA2-
EV01**



Isidro J Gallardo Navarro

**Ficha:3070299
2025**

**Tecnología en Análisis y Desarrollo de
Software.**

ADSO

Título: El Tablero y la Nomenclatura en Ajedrez Según Grau

De: Aprendiz Digital I J Gallardo Navarro

Programa: Análisis y Desarrollo de Software – ADSO, ficha: 3070299

Palabras Clave: Ajedrez, Notación, Sistema algebraico, Notación simplificada, Lenguajes de programación, Python, C, Abstracción, Minimalismo, Algoritmo, Evolución, Sintaxis, Rendimiento, Casillas, Colores, Dualidad

Introducción

El ajedrez es un juego tan antiguo como enigmático el cual ha acompañado a la humanidad durante su desarrollo tecnológico, el entender el juego y sus principios brinda al practicante habilidades que puede homologar en otras áreas de la vida cotidiana u académica

En el presente escrito se abordará el análisis sobre nociones básicas en del ajedrez, a lo que R. Grau llama rudimentos, estos principios se exponen en el primer tomo y primer capítulo de su magistral tratado sobre el juego ciencia, se abordara el tema de las piezas, el tablero, sus partes, los diferentes tipos de nomenclatura en ajedrez y el uso de estas herramientas en otros idiomas con el fin de poder estudiar partidas históricas e interactuar con la comunidad global ya sea a través del internet de las cosas o torneos presenciales.

El ajedrez se desarrolla en un tablero cuadrado de 64 escaques, R. Grau propone llamarles casillas negras y casillas blancas, una casilla puede entenderse como el resultado de la intersección de una fila con una columna, las cuales son ordenadas en ocho filas que va de manera horizontal y columnas de forma vertical, en la modernidad los tableros de casillas negras han sido remplazados por su connotación en la salud, y se optó por tableros verdes con blanco esto tiene el propósito de no afectar la salud y si bien se sigue usando el término casillas blancas y negras lo adecuado es referirse a casillas claras y casillas oscuras; puesto que estos colores pueden variar, pero el principio de dualidad es inexpugnable del tablero. Con respecto a las partes del tablero podríamos identificar gracias a lo expuesto por Grau; el ala de Dama, ala de Rey, centro, a su vez este último se divide en dos partes centro y gran centro.(Grau, n.d., 25)

DIAGRAMA Nº 46

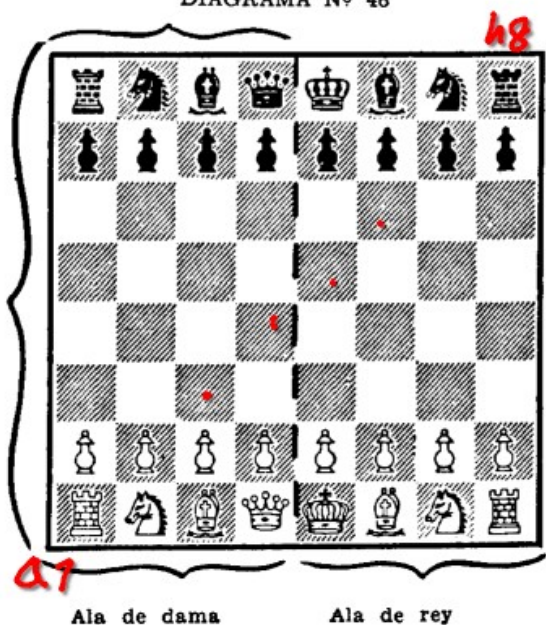
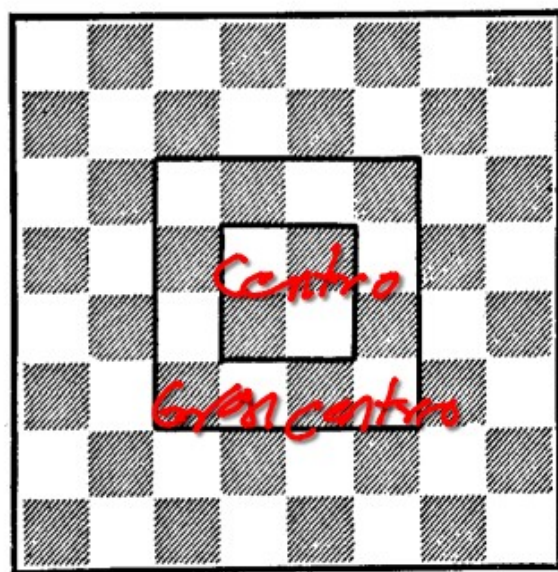
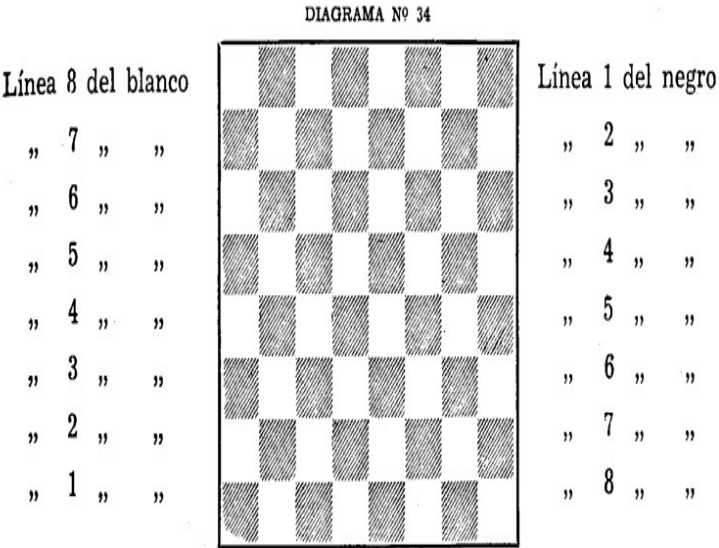
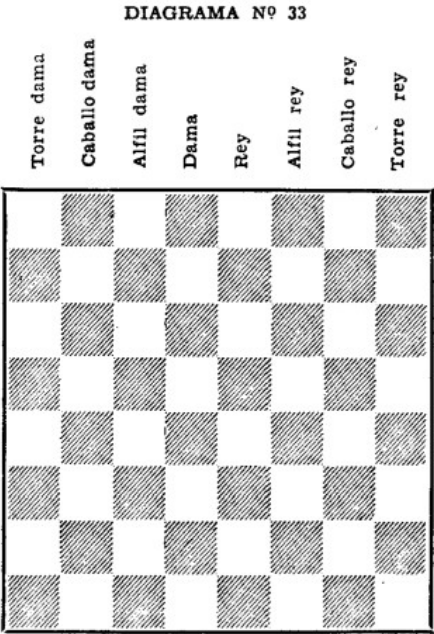


DIAGRAMA Nº 47



El ala de dama comprendería el área del tablero que va desde la torre a la izquierda del tablero hasta la posición de la Dama y el ala del Rey desde la casilla que este ocupa en el centro del tablero hasta la torre de la derecha, por otro lado, el centro y el gran centro del tablero es más sencillo identificarles en nomenclatura algebraica y comprenden las casillas de e4, e5, d4 y d5. El gran centro ocupa el espacio desde la columna; c, d, e y f, desde la tercera fila hasta la sexta. Es importante identificar durante las partidas las piezas que están dentro del centro y gran centro, puesto que es la parte del tablero donde se empiezan las acciones ofensivas y quien domine primero estos escaques es quien goza de iniciativa para el ataque.

Para R. Grau el sistema de nomenclatura descriptivo es el más cómodo para motivo por el cual en su obra se usa este sistema para describir los diagramas y jugadas, también propone este sistema, ya que le da al aprendiz la posibilidad de consultar libros antiguos previos a la invención del sistema algebraico, el sistema descriptivo se basa en el orden inicial de las piezas y es este orden quien da el nombre a las casillas. (Grau, n.d., 24,25)



Como se había mencionado al principio, las casillas son la intersección de una columna y una línea/fila al combinar las designaciones de los diagramas 33 y 34 se obtendría el nombre de las casillas del sistema descriptivo, aclara Grau que existen dos designaciones, como se puede apreciar en el diagrama 34 lo que para blancas es "línea 1" para negras sería "línea 8" pongamos como ejemplo la primera casilla del tablero en sistema algebraico sería "a1" para ambos bandos, en sistema descriptivo para blancas sería "1TD" y para las negras sería "8TD". (Grau, n.d., 25)

DIAGRAMA Nº 35

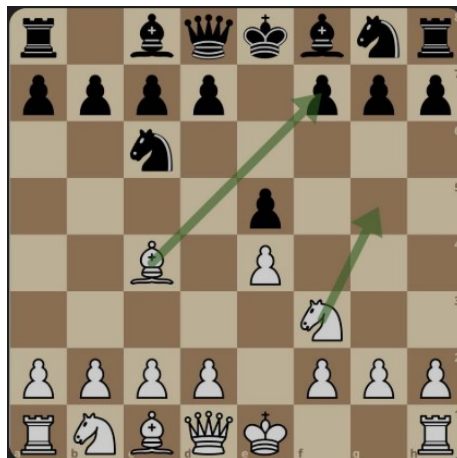
8TD	8CD	8AD	8D	8R	8AR	8CR	8TR
7TD	7CD	7AD	7D	7R	7AR	7CR	7TR
6TD	6CD	6AD	6D	6R	6AR	6CR	6TR
5TD	5CD	5AD	5D	5R	5AR	5CR	5TR
4TD	4CD	4AD	4D	4R	4AR	4CR	4TR
3TD	3CD	3AD	3D	3R	3AR	3CR	3TR
2TD	2CD	2AD	2D	2R	2AR	2CR	2TR
1TD	1CD	1AD	1D	1R	1AR	1CR	1TR

DIAGRAMA Nº 36

1TD	1CD	1AD	1D	1R	1AR	1CR	1TR
2TD	2CD	2AD	2D	2R	2AR	2CR	2TR
3TD	3CD	3AD	3D	3R	3AR	3CR	3TR
4TD	4CD	4AD	4D	4R	4AR	4CR	4TR
5TD	5CD	5AD	5D	5R	5AR	5CR	5TR
6TD	6CD	6AD	6D	6R	6AR	6CR	6TR
7TD	7CD	7AD	7D	7R	7AR	7CR	7TR
8TD	8CD	8AD	8D	8R	8AR	8CR	8TR

Las piezas en el sistema descriptivo son nombradas según el ala del tablero que ocupen por lo que tendríamos "Alfil de Dama" y "Alfil de Rey", con los peones el sistema descriptivo toma como referencia la pieza que está debajo entonces tendríamos "Peón Alfil de Dama" y "Peón Alfil de Rey" a continuación listaré las abreviaciones del sistema descriptivo para las piezas, pero además la de movimientos como captura de pieza, movimiento extraordinarios como enroque corto, largo, jaque y jaque mate:

Ejemplificaré el uso de las siguientes herramientas listadas con la apertura Italiana hasta el tercer movimiento:



Blancas:

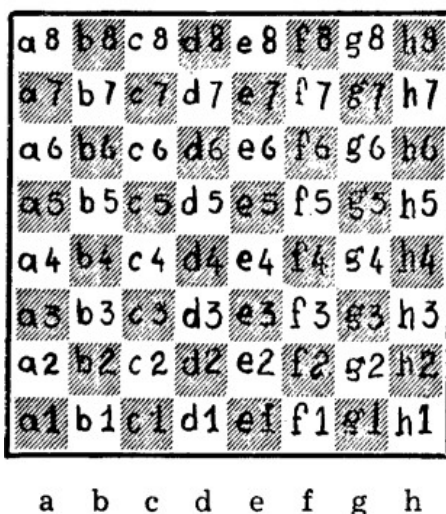
1. P4R (Peón a la cuarta del Rey)
2. C3AR (Caballo a la tercera del Alfil de Rey)
3. 4A (Alfil a la cuarta del Alfil de Dama)

Negras:

1. P4R (Peón a la cuarta del Rey)
2. C3AD (Caballo a la tercera del Alfil de Dama)

El sistema algebraico es descrito por Grau brevemente, puesto que en su obra el emplea el método descriptivo continúa explicando la nomenclatura de las casillas, explica que las líneas/filas van de la 1 a 8 donde las piezas blancas ocupan las líneas 1 y 2 y las piezas negras de la 7 y 8 y las columnas van desde la "a" hasta la "h" se puede apreciar esto en el diagrama 45 de la obra.

DIAGRAMA Nº 45



Las piezas en el sistema algebraico son nombradas por la inicial en mayúscula excepto el peón quien no lleva inicial, aclaremos que las columnas se escriben en minúsculas, para representar un movimiento primero se coloca la inicial de la pieza en cuestión al lado de la casilla que está ocupando seguido de "-" y la casilla a la que se dirige en el tomo se ejemplifica esto de la siguiente manera: "R c8 - d7 " esto quiere indicar que el Rey ocupa la casilla c8 se dirige a d7. Los signos son los mismos del sistema descriptivo; sin embargo, los listaremos igualmente a continuación:

Piezas:

T: Torre

C: Caballo de Dama

D: Dama

R: Rey

A: Alfil

Movimientos:

Captura de Pieza: X

Enroque Corto: 0-0

Enroque Largo: 0-0-0

Jaque: +

Jaque Mate: ++

Buena jugada: !

Genial: !!

Error: ?

Grave: ??

Como podemos apreciar ya no se tomaría el ala donde inicia la pieza para la nomenclatura del mismo aun así se desarrolló un sistema simplificado en donde se omite el origen de casilla de la pieza y solo se anota a donde se dirige, en el anterior ejemplo veíamos que el Rey en c8 se dirige a d7, en este sistema simplificado solo se anotaría "Rd7" este sistema simplificado es el más se utiliza en la actualidad siendo la opción por defecto en plataformas como chess.com, en el sistema algebraico simplificado solo se hace referencia a la casilla de origen cuando hay dos piezas que puedan ir al mismo lugar, veamos ahora como se vería la apertura Italiana con este sistema:

1. e4 - e5

2. Cf3 - Cc6

3. Ac4 -

Como se puede apreciar la evolución de la notación ha ido hacia el minimalismo, una de las cosas que no se puede hacer con el sistema descriptivo que si podemos hacer con el sistema algebraico es calcular el color de la casilla sin ver el tablero, esto con el fin de saber a que casillas pueden ir nuestras piezas y como propone Grau, el desarrollo del órgano ajedrecístico de la visión mediata, para este propósito se tendría que desarrollar un algoritmo básico el cual nos permita calcular el color de las casillas sin ver el tablero. (Alvarez Caules, n.d.)

Sabemos que el tablero está constituido por ocho filas y ocho columnas, las filas se les ha asignado números y también hemos expuesto que las casillas es la intersección de una fila con una columna, pero a las columnas letras, sumar letras y número es un poco abstracto y necesitamos algo sencillo que podamos ejecutar mentalmente de manera ágil por lo cual propongo homologar las letras a números; a=1, b=2, c=3, d=4, e=5, f=6 g=7, h=8 esto nos simplificaría nuestro ejercicio que constaría en sumar las coordenadas si el resultado la suma de la fila y la columna es un número par el escaque es oscuro si es impar es claro hallemos el color de la casilla

f6:

f = 6

6 + 6 = 12 par

La casilla es oscura.

Se puede entender que la evolución de la sintaxis de notación en ajedrez ha evolucionado de manera que cada vez se ha buscado ser más sencilla, haciendo un símil con el campo de la programación podríamos compararle con los lenguajes de alto nivel y bajo nivel donde bajo nivel se refiere que el lenguaje está cerca a la comprensión de las máquinas y alto nivel a la sintaxis del lenguaje que usamos los humanos cotidianamente, ejemplo de lenguajes bajo nivel podríamos mencionar a "C" y de alto nivel Python.

La comparación del lenguaje C y Python es preciso en este caso porque siendo Python un lenguaje de alto nivel busca tener el rendimiento de un lenguaje de bajo nivel simplificando su sintaxis, también podríamos ver algo similar con el lenguaje Java quien toma como referencia a C en su sintaxis, esto sería igual a lo que paso con la notación algebraica al inspirar la creación de la notación simplificada en ajedrez, para ilustrar lo anterior se desarrollara un programa en lenguaje C y en Python basados en el algoritmo para identificar el color de la casilla que desarrollamos anteriormente y así podríamos identificar las diferencias y particularidades de cada lenguaje. (Trigo Aranda, n.d., 89)

```
1 #include <stdio.h>
2 #include <ctype.h>
3 #include <string.h>
4
5 int main() {
6     char casilla[3];
7     char columna;
8     int fila;
9     int valor_columna, suma;
10
11     printf("=== DETERMINADOR DE COLOR DE CASILLAS DE AJEDREZ ===\n\n");
12     printf("Ingrese la casilla (ejemplo: f6, a1, h8): ");
13
14     scanf("%2s", casilla);
15
16     columna = tolower(casilla[0]);
17     fila = casilla[1] - '0';
18
19     if (columna < 'a' || columna > 'h' || fila < 1 || fila > 8) {
20         printf("Error: Casilla inválida. Use formato como a1, f6, h8, etc.\n");
21         return 1;
22     }
23
24     valor_columna = columna - 'a' + 1;
25     suma = valor_columna + fila;
26
27     printf("\nAnálisis de la casilla %c%d:\n", columna, fila);
28     printf("Columna '%c' = %d\n", columna, valor_columna);
29     printf("Fila = %d\n", fila);
30     printf("%d + %d = %d", valor_columna, fila, suma);
31
32     if (suma % 2 == 0) {
33         printf(" (par)\n");
34         printf("La casilla %c%d es OSCURA.\n", columna, fila);
35     } else {
36         printf(" (impar)\n");
37         printf("La casilla %c%d es CLARA.\n", columna, fila);
38     }
39
40     printf("\n¿Desea probar otra casilla? (s/n): ");
41     char respuesta;
42     scanf(" %c", &respuesta);
43
44     if (respuesta == 's' || respuesta == 'S') {
45         main();
46     }
47
48     return 0;
49 }
```

Para ser breves en nuestro escrito nos referiremos a los rasgos más llamativos del código en C, por ejemplo el hecho de que se importen librerías al inicio del código para imprimir mensajes por

consola, cosa que en lenguajes de alto nivel daríamos por hecho que vienen integradas, otro detalle de este código es que necesita un paso adicional el cual se le llama compilar y se ejecutaría con el siguiente script **«gcc index.c -o index»** una vez compilado nuestro código se genera un binario llamado **“index”** el cual ejecutaría simplemente llamándole por consola ./index.

Es turno de nuestro código en Python; a diferencia del lenguaje **C** podemos apreciar no requerimos importar librerías para imprimir por consola, la palabra reservada para imprimir **“print”** es de fácil recordar, pues se escribe igual que en el idioma inglés, no requerimos especificar los tipos de datos de nuestra variable y para ejecutar el código no hace falta compilar, ya que es un lenguaje interpretado; dicho intérprete está escrito en C lo cual corrobora nuestra hipótesis de la tendencia a simplificar las sintaxis de los lenguajes de programación. Para ejecutar nuestro código en Python haría falta el siguiente script «py ./index.py» o «python3 ./index.py». (Ramirez Hernandez, 2000, 52)

```
def main():
    """
    Determina el color de una casilla de ajedrez basándose en la suma
    del valor numérico de la columna y la fila.
    """
    print("=== DETERMINADOR DE COLOR DE CASILLAS DE AJEDREZ ===")

    while True:
        casilla = input("Ingrese la casilla (ejemplo: f6, a1, h8): ").strip().lower()

        if len(casilla) != 2:
            print("Error: Formato inválido. Use formato como a1, f6, h8, etc.\n")
            continue

        columna = casilla[0]
        fila_char = casilla[1]

        if columna < 'a' or columna > 'h':
            print("Error: La columna debe estar entre 'a' y 'h'.\n")
            continue

        if not fila_char.isdigit() or int(fila_char) < 1 or int(fila_char) > 8:
            print("Error: La fila debe estar entre 1 y 8.\n")
            continue

        fila = int(fila_char)
        valor_columna = ord(columna) - ord('a') + 1
        suma = valor_columna + fila

        print(f"Análisis de la casilla {columna}{fila}:")
        print(f"Columna '{columna}' = {valor_columna}")
        print(f"Fila '{fila}' = {fila}")
        print(f"valor_columna + {fila} = {suma}, end=")

        if suma % 2 == 0:
            print(" (par)")
            print(f"La casilla {columna}{fila} es OSCURA.")
        else:
            print(" (impar)")
            print(f"La casilla {columna}{fila} es CLARA.")

        print("\n¿Desea probar otra casilla? (s/n): ", end="")
        respuesta = input().strip().lower()

        if respuesta not in ['s', 'si', 'sí', 'y', 'yes']:
            break

    print()

if __name__ == "__main__":
    main()
```

Abordamos la relación entre los lenguajes de programación de bajo nivel como C y de alto nivel como Python. Así como Python simplifica la sintaxis de C manteniendo su rendimiento, la notación simplificada del ajedrez optimiza la comunicación sin perder la precisión. Ambos casos demuestran una tendencia en la tecnología y la lógica humana: buscar la eficiencia a través de la abstracción, permitiendo a los usuarios centrarse en la lógica del problema (la partida de ajedrez o el algoritmo) en lugar de los detalles de la implementación (la notación completa o la gestión manual de la memoria).

References

Alvarez Caules, C. (n.d.). Arquitectura Java - Blog sobre Java EE. Retrieved August 19, 2025, from

<http://www.arquitecturajava.com>

Grau, R. (n.d.). *Tratado general de ajedrez*.

Ramírez Hernández, A. (2000). *Antología de lenguajes de programación*.

Trigo Aranda, V. (n.d.). *Historia y evolución de los lenguajes de programación*.