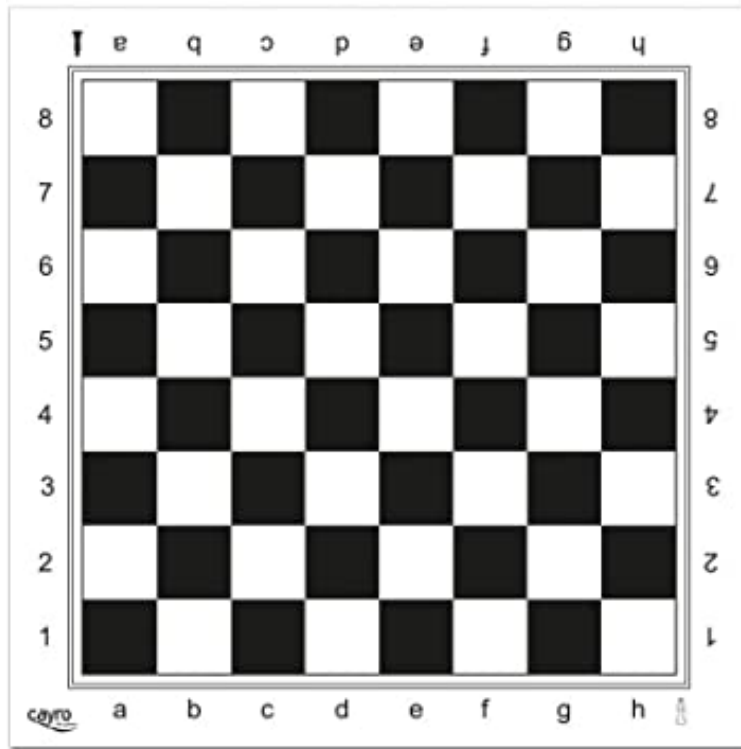


Proyecto 1 Módulo 3 *Tablero de Ajedrez*

En este proyecto se debe recrear (hasta donde se pueda) un tablero de ajedrez.



Este se debe de recrear con ayuda del lenguaje de programación Python y la librería [Turtle](#) y debe de contener las características que se describen a continuación

Especificaciones Generales

Formato de entrega, se debe entregar 2 archivos:

1. **Código del programa:** Guardar el código con el nombre *proyecto-tablero.py*
2. **Reporte del código:** Este debe conllevar los siguientes puntos:
 1. Nombre del alumno
 2. Fecha
 3. Nombre del proyecto
 4. Problemática que se soluciona
 5. Código realizado por el alumno

```
"""
@author: nombre de quien hace el programa
@date: Fecha en la que se realiza el programa
@description: Descripcion del programa
"""

Completar el código de la actividad que se realiza
```

6. Captura de pantalla del código ejecutandose
7. Captura de pantalla del validador (En este proyecto no hay validador, este punto se omite)

Especificaciones del código

A continuación se describe las características que debe contener el código y el dibujo

1. Función creadora de la figura cuadrado, se recomienda que este cuadrado tenga de tamaño 50 a 70, **no se recomienda hacerlo de 100**, la función debe llamarse *cuadrado*

```
##Funcion creadora de cuadrado
def cuadrado(tam):
    for _ in range(4):
        frank.forward(tam)
        frank.right(90)
```

2. Función creadora de una hilera horizontal de cuadros de ajedrez, es decir una hilera de 8 cuadros de ajedrez, esta función debe llamarse *hilera_horizontal*. **OJO: El código está incompleto**

```
###Codigo correspondiente a la hilera horizontal
def hilera_horizontal(tam):
    for i in range(4):
        ##Cuadro 1
        frank.color("black", lista[0])
        frank.begin_fill()
        cuadrado(tam)
        frank.fd(tam)
        frank.end_fill()
```

1. **Opcional:** Lograr que se pinten los cuadros de color (éstos pueden ser blanco y negro o bien azul y gris). **TIP: Se recomienda crear una lista con los colores disponibles y generar 2 opciones para cuadrados**

```
##OJO: El codigo es una recomendacion
color1=["black", "white"]
color2=["white", "black"]
```

```

lista= color2
if opcion==1:
    lista = color1
#####Cuadro 1
#Completar código
###Cuadro 2
frank.color("black", lista[1])
frank.begin_fill()
cuadrado(tam)
frank.fd(tam)
frank.end_fill()

```

3. Repetición de 8 veces de la función creadora de línea horizontal, esta función debe poder llenar el tablero de 8x8. **TIP: Se recomienda posicionarse en la posición (-250,300), esto se logra con el comando goto(), así como moverse con los comandos penup() y pendown()**

```

###OJO, el código está incompleto y se debe completar conforme a la realización del
proyecto
def tablero(tam):
    frank.penup()
    frank.goto(-250,300)
    frank.pendown()
    for i in range(1,9):
        hilera_horizontal(tam)
        frank.penup()
        frank.rt(90)
        frank.fd(tam)
        frank.rt(90)
        frank.fd(tam*8)
        frank.lt(180)
        frank.pendown()

```

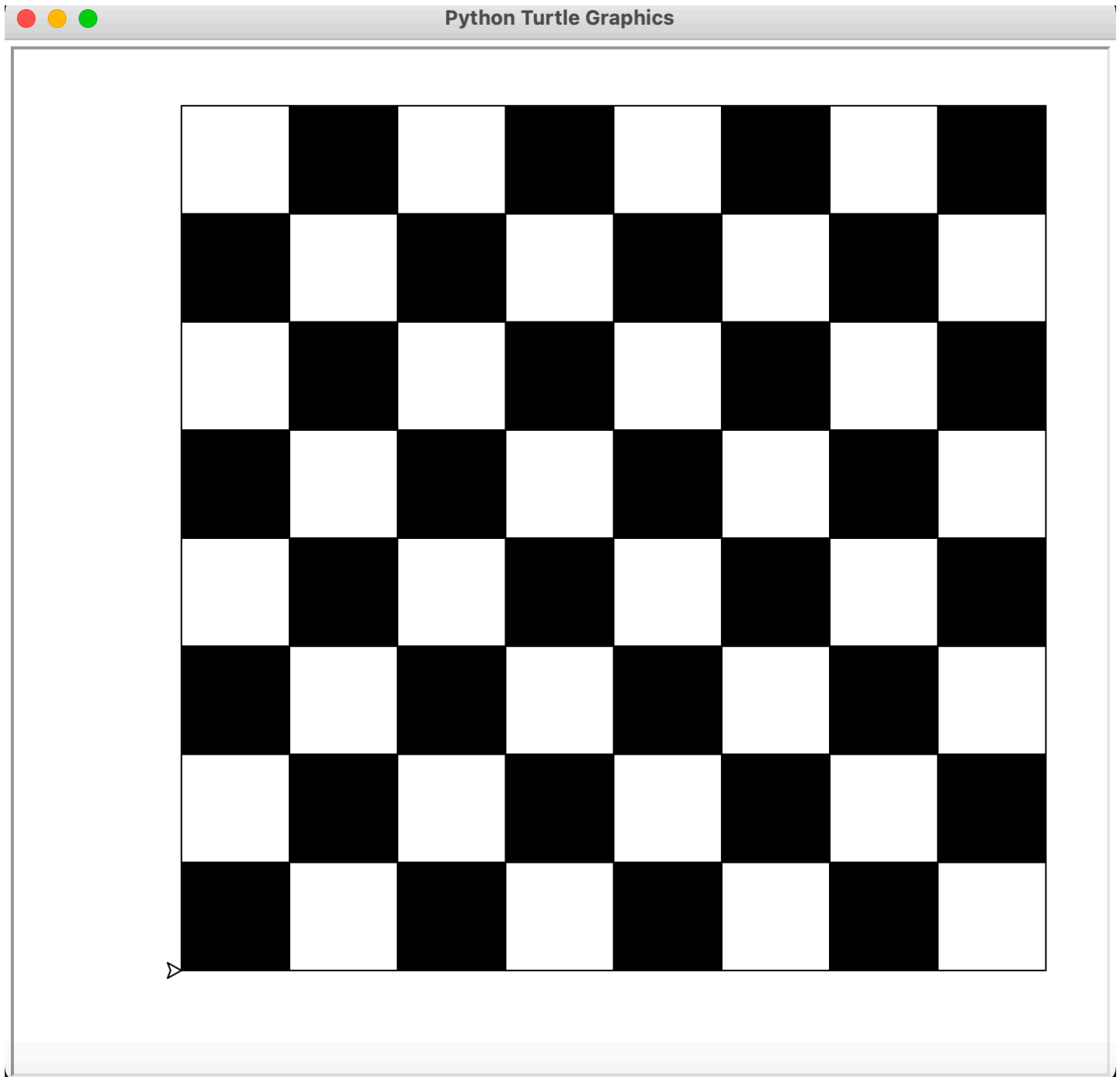
4. Por último debe haber una función llamado main(), la cual debe activar la realización de todo el código

```

def main(tam1):
    ##Dibujar el tablero
    tablero(tam1)
    turtle.done()
main(70)

```

Ejecución y demostración del programa



Extras

Esta sección detalla los puntos extra que el programa puede tener

- **Color de pantalla de fondo:** Proponer e implementar un color diferente a la pantalla blanca. *TIP: Usar el `screen de turtle`*
- **Color de Tortuga y forma de la tortuga:** Proponer un color y forma de la tortuga
- **Líneas externas y letras del tablero:** Generar letras y números correspondientes a las posiciones exteriores del tablero. A continuación se presenta el resultado de éste punto
 - **TIP:** Se recomienda usar la función *write* y definir una lista con los datos que se escribieran

