

Informe de Laboratorio 04

Tema: Python

Nota

Estudiante	Escuela	Asignatura
Rutbel Carlos Ttito Campos rttitoca@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Programación Web 2 Semestre: III Código: 20231001

Laboratorio	Tema	Duración
04	Python	04 horas



Semestre académico	Fecha de inicio	Fecha de entrega
2023 - A	Del 10 Abril 2023	Al 17 Abril 2023

1. Equipos, materiales y temas utilizados

- Sistema Operativo Ubuntu 22.04.2 LTS
- Git 2.39.2.
- Cuenta en GitHub con el correo institucional.
- Python
- Arreglos en python

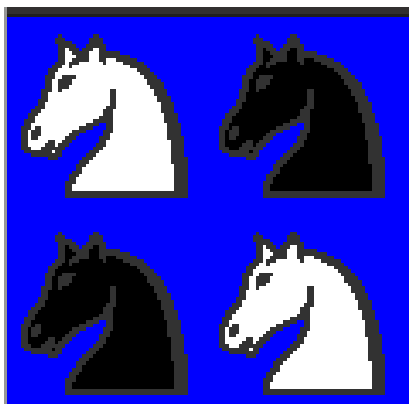
2. Tarea

- URL GitHub de Tarea del Ajedrez <https://github.com/rescobedoq/pw2/tree/main/labs/lab04/Tarea-del-Ajedrez>
- En esta tarea usted pondrá en práctica sus conocimientos de programación en Python para dibujar un tablero de Ajedrez.
- La parte gráfica ya está programada, usted sólo tendrá que concentrarse en las estructuras de datos subyacentes.
- Con el código proporcionado usted dispondrá de varios objetos de tipo Picture para poder realizar su tarea:

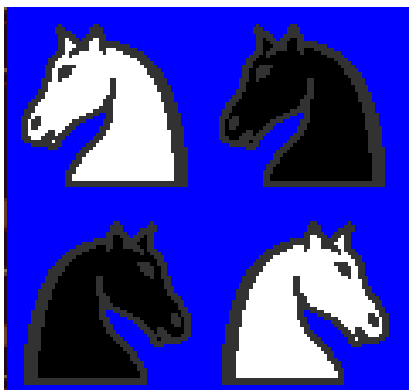
	rock
	knight
	bishop
	queen
	king
	square

■ Ejercicios:

- Para resolver los siguientes ejercicios sólo está permitido usar ciclos, condicionales, definición de listas por comprensión, sublistas, map, join, (+), lambda, zip, append, pop, range.
- Implemente los métodos de la clase Picture. Se recomienda que implemente la clase picture por etapas, probando realizar los dibujos que se muestran en la siguiente preguntas.
- Usando únicamente los métodos de los objetos de la clase Picture dibuje las siguientes figuras (invoque a draw):
 - ejercicio a



- ejercicio b



o ejercicio c



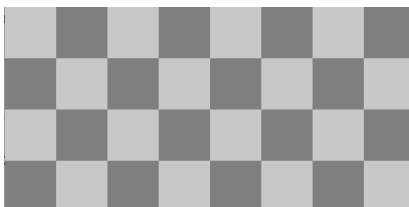
o ejercicio d



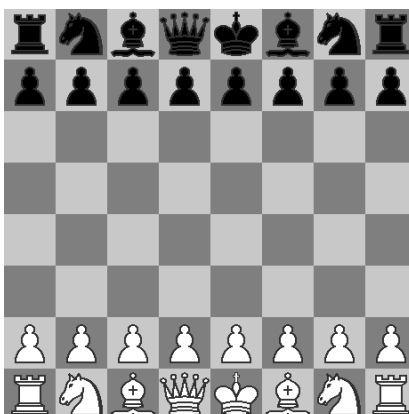
o ejercicio e



o ejercicio f



o ejercicio g



3. Solucion de Ejercicios

3.1. Creando entorno virtual

```
* carlos@core:~/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez$ virtualenv -p python3 .
created virtual environment CPython3.10.6.final.0-64 in 228ms
creator CPython3Posix(dest=/home/carlos/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez, clear=False, no_vcs_ignore=False,
global=False)
seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy, app_data_dir=/home/carlos/.local/share/virtualenv)
added seed packages: pip==22.0.2, setuptools==59.6.0, wheel==0.37.1
activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator
```

3.2. Instalando Pygame

```
(tareaAjedrez) carlos@core:~/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez$ source bin/activate
(tareaAjedrez) carlos@core:~/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez$ pip freeze
(tareaAjedrez) carlos@core:~/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez$ pip --version
pip 22.0.2 from /home/carlos/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez/lib/python3.10/site-packages/pip (python 3.10)
(tareaAjedrez) carlos@core:~/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez$ pip install pygame
Collecting pygame
Using cached pygame-2.4.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (13.8 MB)
Installing collected packages: pygame
Successfully installed pygame-2.4.0
(tareaAjedrez) carlos@core:~/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez$ pip freeze
pygame==2.4.0
(tareaAjedrez) carlos@core:~/Documentos/uns2023/pw2-lab-d-23a/Lab4/tareaAjedrez$ deactivate
```

3.3. Funciones de la clase Picture

Listing 1: picture.py

```
1 from colors import *
2 class Picture:
3     def __init__(self, img):
4         self.img = img
5
6     def __eq__(self, other):
7         return self.img == other.img
8
9     def _invColor(self, color):
10        if color not in inverter:
11            return color
12        return inverter[color]
13
14    def verticalMirror(self):
15        """ Devuelve el espejo vertical de la imagen """
16        vertical = []
17        cadAux = ""
18        for i in range(0, len(self.img)):
19            for j in range(0, len(self.img[i])):
20                cadAux = self.img[i][j] + cadAux;
21            vertical.append(cadAux)
22            cadAux = ""
23
24        return Picture(vertical)
25
26    def horizontalMirror(self):
27        """ Devuelve el espejo horizontal de la imagen """
28        horizontal = []
29
30        for i in range(len(self.img)-1,-1,-1):
31            horizontal.append(self.img[i])
```

```
32
33     return Picture(horizontal)
34
35 def negative(self):
36     """ Devuelve un negativo de la imagen """
37     negative = []
38     aux = ""
39     for i in range(0, len(self.img)):
40         for j in range(0, len(self.img[i])):
41             aux += self._invColor(self.img[i][j])
42         negative.append(aux)
43     aux = ""
```

3.4. Ejercicios

Listing 2: Ejercicio2a.py

```
1 from interpreter import draw
2 from chessPictures import *
3 #Par de caballos blanco y negro
4 CaballosA = knight.join(knight.negative())
5 #Par de caballos negro y blanco
6 CaballosB = CaballosA.negative()
7 #dibujando grupo de caballos
8 grupoCaballos = Picture(CaballosA.img + CaballosB.img)
9 draw(grupoCaballos)
```

Listing 3: Ejercicio2b.py

```
1 from interpreter import draw
2 from chessPictures import *
3
4 CaballosA = knight.join(knight.negative())
5 CaballosB = CaballosA.verticalMirror()
6
7 grupoCaballos = Picture(CaballosA.img + CaballosB.img)
8 draw(grupoCaballos)
```

Listing 4: Ejercicio2c.py

```
1 from interpreter import draw
2 from chessPictures import *
3
4 grupoReinas = queen.horizontalRepeat(4)
5 draw(grupoReinas)
```

Listing 5: Ejercicio2d.py

```
1 from interpreter import draw
2 from chessPictures import *
3
4 filaCuadrosA = (square.join(square.negative())).horizontalRepeat(4)
5 draw(filaCuadrosA)
```

Listing 6: Ejercicio2e.py

```
1 from interpreter import draw
2 from chessPictures import *
3
4 filaCuadrosB = ((square.negative()).join(square)).horizontalRepeat(4)
5
6 draw(filaCuadrosB)
```

Listing 7: Ejercicio2f.py

```
1 from interpreter import draw
2 from chessPictures import *
3
4 filaCuadrosA = (square.join(square.negative())).horizontalRepeat(4)
5 filaCuadrosB = filaCuadrosA.negative()
6 filaCuadrosAB = Picture(filaCuadrosA.img + filaCuadrosB.img)
7 grupoCuadros = filaCuadrosAB.verticalRepeat(2)
8 draw(grupoCuadros)
```

Listing 8: Ejercicio2g.py

```
1 from interpreter import draw
2 from chessPictures import *
3 #estructura de piezas grandes
4 izq = (rock.join(knight)).join(bishop)
5 centro = queen.join(king)
6 der = (bishop.join(knight)).join(rock)
7
8 #estructura de filas de cuadros
9 filaCuadrosA = (square.join(square.negative())).horizontalRepeat(4)
10 filaCuadrosB = filaCuadrosA.negative()
11
12 peones = pawn.horizontalRepeat(8)
13 piezas = (izq.join(centro)).join(der)
14
15 blancas = Picture(peones.img + piezas.img)
16 negras = Picture(piezas.img + peones.img).negative()
17
18 filaCuadrosAB = Picture(filaCuadrosA.img + filaCuadrosB.img)
19 #grupos de piezas
20 grupoBlancas = blancas.up(filaCuadrosAB)
21 grupoNegras = (negras.up(filaCuadrosAB))
22 grupoCuadros = filaCuadrosAB.verticalRepeat(2)
23 #dibujando tablero
24 tablero = Picture(grupoNegras.img + grupoCuadros.img + grupoBlancas.img)
25 draw(tablero)
```

3.5. Capturas de Ejecucion



Figura 1: ejecucion ejercicio a



Figura 2: ejecucion ejercicio b

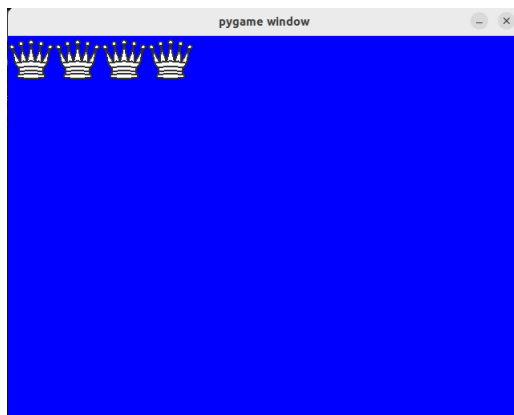


Figura 3: ejecucion ejercicio c

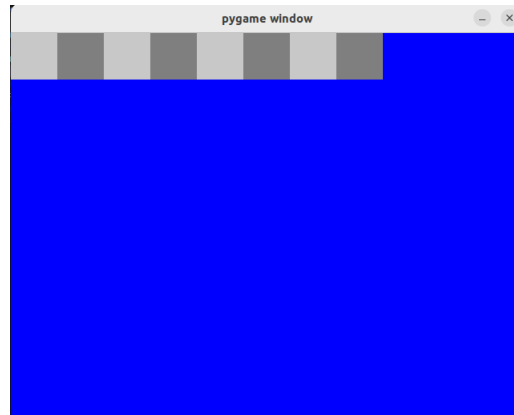


Figura 4: ejecucion ejercicio d

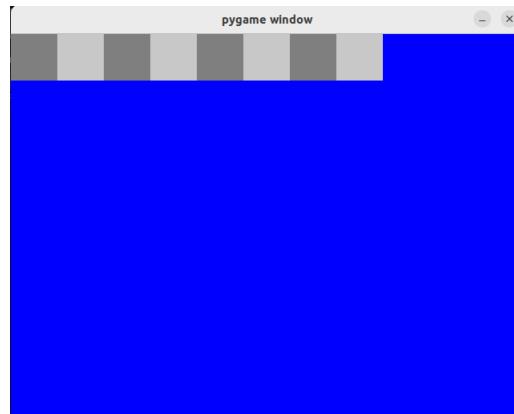


Figura 5: ejecucion ejercicio e

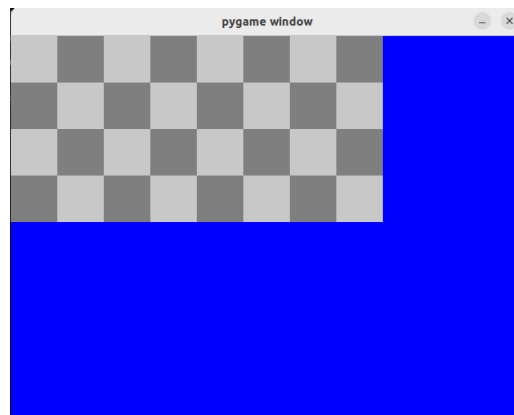


Figura 6: ejecucion ejercicio f

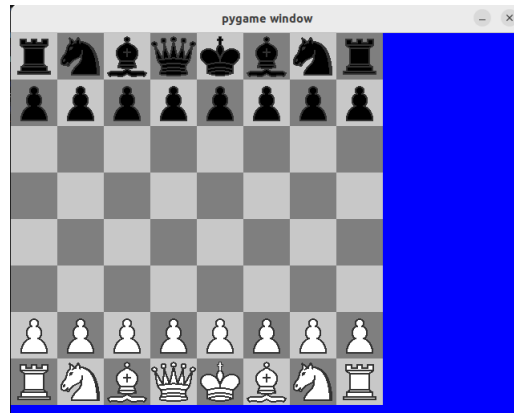


Figura 7: ejecucion ejercicio g

4. URL de Repositorio Github

- URL del Repositorio GitHub para clonar o recuperar:
- <https://github.com/RutbelCarlosTC/pw2-lab-d-23a.git>
- URL para el laboratorio 04 en el Repositorio GitHub.
- <https://github.com/RutbelCarlosTC/pw2-lab-d-23a/tree/main/Lab4>

5. Actividades con el repositorio GitHub

5.1. Commits

- Primer commit agregando carpeta con plantillas de ejercicios de Tarea-del-Ajedrez

```
commit c64241c2e28cbd06a6d32ca1e41bba1225caf46f
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:02:05 2023 -0500

se agrego lab4 con carpeta inicial TareaAjedrez
```

- Agregando archivo .gitignore

```
commit 8366d3d2ea1f27e96ce4f63b61d19fd01a662cbf
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:41:13 2023 -0500

se agrego git ignore
```

- Completando funciones

```
commit 5ac0e307b638f8921fa83a9f4febbff6cd32a276
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:44:18 2023 -0500

se completo funcion verticalMirror
```

(a) verticalMirror

```
commit 523274531f20ca1207c52d74cda3a5a71fa1ad90
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:44:58 2023 -0500

se completo funcion horizontalMirror
```

(b) horizontalMirror

```
commit a4efe68ef8bdaca6ce0837955840342c3a26dbac
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:46:06 2023 -0500

se completo funcion join
```

(c) join

```
commit aef37f7740bbe3d51740078238b0fe733a83506
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:45:31 2023 -0500

se completo funcion negative
```

(d) Negative

```
commit 7d40554199374f14eff98d685e5868820d8fa03b
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:48:09 2023 -0500

se completo funcion horizontalRepeat
```

(e) horizontalRepeat

```
commit 3e8d2f16d8821642a1e6b8ffd0a2fc87235bc28a
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:48:39 2023 -0500

se completo funcion verticalRepeat
```

(f) verticalRepeat

```
commit 7d40554199374f14eff98d685e5868820d8fa03b
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:48:09 2023 -0500

se completo funcion horizontalRepeat
```

(g) horizontalRepeat

```
commit 3e8d2f16d8821642a1e6b8ffd0a2fc87235bc28a
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 18:48:39 2023 -0500

se completo funcion verticalRepeat
```

(h) verticalRepeat

- Completando ejercicios:

```
commit d4ab22a4bb3c9192f8b01a7727afebbd32488d87
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 19:02:30 2023 -0500

se completo ejercicio a
```

(a) Commit Ejercicio a

```
commit 1afaf3c3232a6330f91a95ddc209cfd6be279adb
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 19:03:00 2023 -0500

se completo ejercicio b
```

(b) Commit Ejercicio b

```
commit 6365566a08652b456bb22f1e357dc5d9c345544e
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 19:03:35 2023 -0500

se completo ejercicio c
```

(c) Commit Ejercicio c

```
commit 2e42f1276b55aec715fd0fe052ea8c23c4fef084
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 19:04:31 2023 -0500

se completo ejercicio d fila de cuadros
```

(d) Commit Ejercicio d

```
commit 970d5b8640d914a48dc66599277f6ab0f5fe8bf9
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 19:04:54 2023 -0500

se completo ejercicio e fila de cuadros
```

(e) Commit Ejercicio e

```
commit 087455eb3e36f8703c6917914aff533a6e2e83ff
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 19:05:16 2023 -0500

se completo ejercicio f grupo de cuadros
```

(f) Commit Ejercicio f

```
...skipping...
commit b66a8b6b05afa252beea6985ad10b8ccc71618ea
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sat Jun 3 19:07:07 2023 -0500

se completo ejercicio g tablero de ajedrez
```

(g) Commit Ejercicio g

6. Pregunta: ¿Para qué sirve el directorio pycache?

Se utiliza para almacenar los archivos de código compilado en formato de bytecode de Python (*.pyc) para mejorar el rendimiento en futuras ejecuciones.

7. Rúbricas

7.1. Entregable Informe

Tabla 1: Tipo de Informe

Informe	
Latex	El informe está en formato PDF desde Latex, con un formato limpio (buena presentación) y facil de leer.

7.2. Rúbrica para el contenido del Informe y demostración

- El alumno debe marcar o dejar en blanco en celdas de la columna **Checklist** si cumple con el ítem correspondiente.
- Si un alumno supera la fecha de entrega, su calificación será sobre la nota mínima aprobada, siempre y cuando cumpla con todos los ítems.
- El alumno debe autocalificarse en la columna **Estudiante** de acuerdo a la siguiente tabla:

Tabla 2: Niveles de desempeño

	Nivel			
Puntos	Insatisfactorio 25 %	En Proceso 50 %	Satisfactorio 75 %	Sobresaliente 100 %
2.0	0.5	1.0	1.5	2.0
4.0	1.0	2.0	3.0	4.0

Tabla 3: Rúbrica para contenido del Informe y demostración

Contenido y demostración		Puntos	Checklist	Estudiante	Profesor
1. GitHub	Hay enlace URL activo del directorio para el laboratorio hacia su repositorio GitHub con código fuente terminado y fácil de revisar.	2	X	2	
2. Commits	Hay capturas de pantalla de los commits más importantes con sus explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).	4	X	3	
3. Código fuente	Hay porciones de código fuente importantes con numeración y explicaciones detalladas de sus funciones.	2	X	2	
4. Ejecución	Se incluyen ejecuciones/pruebas del código fuente explicadas gradualmente.	2	X	2	
5. Pregunta	Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).	2	X	2	
6. Fechas	Las fechas de modificación del código fuente están dentro de los plazos de fecha de entrega establecidos.	2	X	2	
7. Ortografía	El documento no muestra errores ortográficos.	2	X	1	
8. Madurez	El Informe muestra de manera general una evolución de la madurez del código fuente, explicaciones puntuales pero precisas y un acabado impecable. (El profesor puede preguntar para refrendar calificación).	4	X	3	
Total		20		17	

8. Referencias

- <https://www.w3schools.com/java/default.asp>
- <https://www.geeksforgeeks.org/insertion-sort/>