
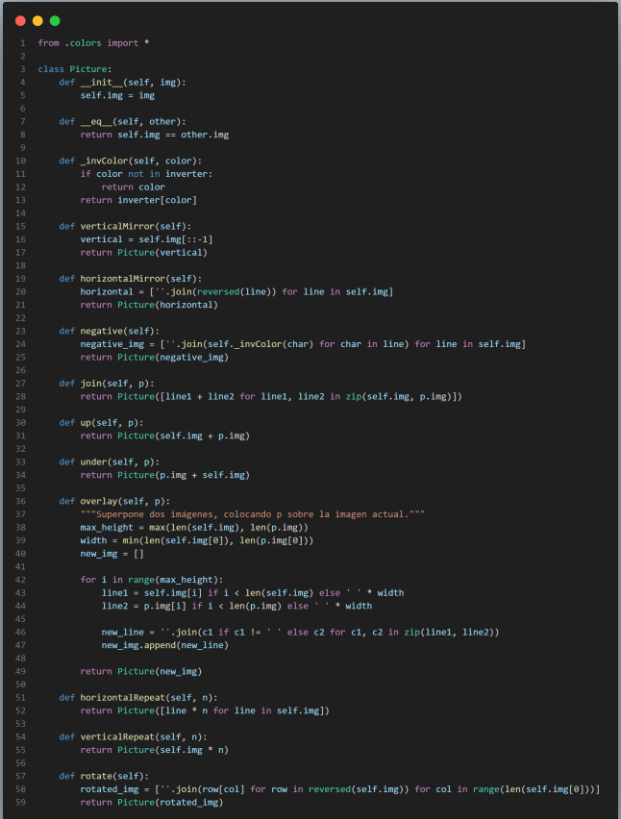
	<p style="text-align: center;"><b>UNIVERSIDAD NACIONAL DE SAN AGUSTIN</b>  <b>FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS</b>  <b>ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</b></p>	
<p style="text-align: center;"><b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p><b>Aprobación:</b> 2022/03/01</p>	<p><b>Código:</b> GUIA-PRLE-001</p>	<p><b>Página:</b> 1</p>

## INFORME DE LABORATORIO

INFORMACIÓN BÁSICA					
<b>ASIGNATURA:</b>	<b>PROGRAMACIÓN WEB 2 – Grupo B</b>				
<b>TÍTULO DE LA PRÁCTICA:</b>					
<b>NÚMERO DE PRÁCTICA:</b>	<b>04</b>	<b>AÑO LECTIVO:</b>	<b>2024 – B</b>	<b>NRO. SEMESTRE:</b>	<b>03</b>
<b>FECHA DE PRESENTACIÓN</b>	<b>18/10/2024</b>	<b>HORA DE PRESENTACIÓN</b>	<b>20:00</b>		
<b>INTEGRANTE (s)</b> <b>NIKOLE VALERY SALAS IDME</b>				<b>NOTA (0-20)</b>	
<b>DOCENTE(s):</b> <ul style="list-style-type: none"> <li><b>Mg. Ing. Lino José Pinto Oppe</b></li> </ul>					

RESULTADOS Y PRUEBAS
<p><b>I. EJERCICIOS RESUELTOS:</b></p> <p>1. 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.</p> <div data-bbox="459 1294 1129 2145">  <pre> 1 from .colors import * 2 3 class Picture: 4     def __init__(self, img): 5         self.img = img 6 7     def __eq__(self, other): 8         return self.img == other.img 9 10    def _invertColor(self, color): 11        if color not in inverter: 12            return color 13        return inverter[color] 14 15    def verticalMirror(self): 16        vertical = self.img[::-1] 17        return Picture(vertical) 18 19    def horizontalMirror(self): 20        horizontal = [''.join(reversed(line)) for line in self.img] 21        return Picture(horizontal) 22 23    def negative(self): 24        negative_img = [''.join(self._invertColor(char) for char in line) for line in self.img] 25        return Picture(negative_img) 26 27    def join(self, p): 28        return Picture([line1 + line2 for line1, line2 in zip(self.img, p.img)]) 29 30    def up(self, p): 31        return Picture(self.img + p.img) 32 33    def under(self, p): 34        return Picture(p.img + self.img) 35 36    def overlay(self, p): 37        """Superpone dos imágenes, colocando p sobre la imagen actual.""" 38        max_height = max(len(self.img), len(p.img)) 39        width = min(len(self.img[0]), len(p.img[0])) 40        new_img = [] 41 42        for i in range(max_height): 43            line1 = self.img[i] if i &lt; len(self.img) else ' ' * width 44            line2 = p.img[i] if i &lt; len(p.img) else ' ' * width 45 46            new_line = ''.join(c1 if c1 != ' ' else c2 for c1, c2 in zip(line1, line2)) 47            new_img.append(new_line) 48 49        return Picture(new_img) 50 51    def horizontalRepeat(self, n): 52        return Picture([line * n for line in self.img]) 53 54    def verticalRepeat(self, n): 55        return Picture(self.img * n) 56 57    def rotate(self): 58        rotated_img = [''.join(row[col] for row in reversed(self.img)) for col in range(len(self.img[0])] 59        return Picture(rotated_img) </pre> </div>

2. Usando únicamente los métodos de los objetos de la clase Picture dibuje las siguientes figuras (invoque a draw):

```
1 from Modules_Chess.chessPictures import *
2 from Modules_Chess.interpreter import draw
3 a = knight.join(knight.negative())
4 draw(a.up(a.negative()))
```



```
1 from Modules_Chess.chessPictures import *
2 from Modules_Chess.interpreter import draw
3 a = knight.join(knight.negative())
4 draw(a.up(a.horizontalMirror()))
```



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



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



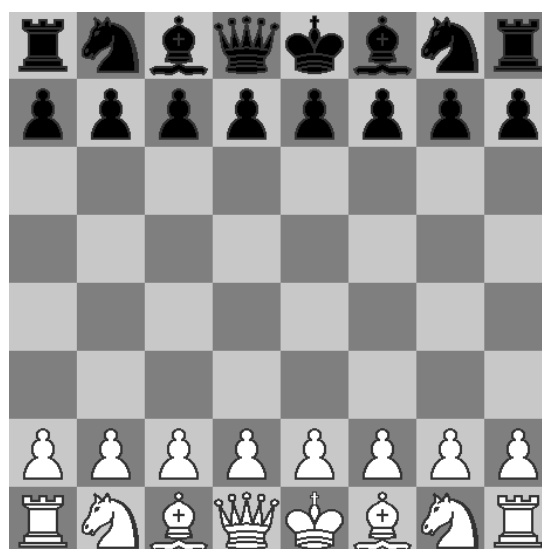
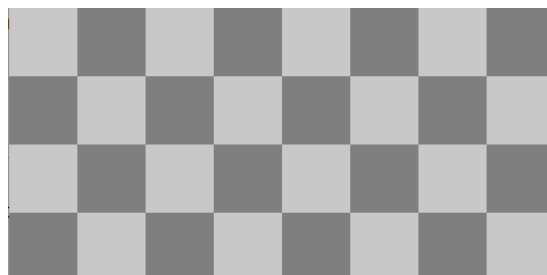
```
1 from Modules_Chess.chessPictures import *
2 from Modules_Chess.interpreter import draw
3 c = (square.negative()).join(square)
4 draw(c.horizontalRepeat(4))
```



```

1 from Modules_Chess.chessPictures import *
2 from Modules_Chess.interpreter import draw
3 d = (square.join(square.negative()).horizontalRepeat(4))
4 draw((d.up(d.negative())).verticalRepeat(2))

```



### 3. COMMITS

end sixth activity	NikoleSalas authored 1 hour ago	Verified	36e1bbd			
end fifth activity	NikoleSalas authored 1 hour ago	Verified	584155f			
end fourth activity	NikoleSalas authored 1 hour ago	Verified	5e3cc3b			
end thirth activity	NikoleSalas authored 1 hour ago	Verified	c68f449			
end second activity	NikoleSalas authored 1 hour ago	Verified	613cac7			
end first exercise	NikoleSalas authored 1 hour ago	Verified	6af53cf			
Complete implementation of Picture and upload of the other modules	NikoleSalas authored 1 hour ago	Verified	dbf5e33			

Tabla 3: Rúbrica para contenido del Informe y evidencias

Contenido y demostración		Puntos	Checklist	Estudiante	Profesor
<b>1. GitHub</b>	Repositorio se pudo clonar y se evidencia la estructura adecuada para revisar los entregables. (Se descontará puntos por error o omisión)	4		2	
<b>2. Commits</b>	Hay porciones de código fuente asociado a los commits planificados con explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).	4		2	
<b>3. Ejecución</b>	Se incluyen comandos para ejecuciones y pruebas del código fuente explicadas gradualmente que permitirían replicar el proyecto. (Se descontará puntos por cada omisión)	4		3	
<b>4. Pregunta</b>	Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).	2		1	
<b>7. Ortografía</b>	El documento no muestra errores ortográficos. (Se descontará puntos por error encontrado)	2		2	
<b>8. Madurez</b>	El Informe muestra de manera general una evolución de la madurez del código fuente con explicaciones puntuales pero precisas, agregando diagramas generados a partir del código fuente y refleja un acabado impecable. (El profesor puede preguntar para refrendar calificación).	4		3	
<b>Total</b>		<b>20</b>		<b>14</b>	