

# Informe de Laboratorio 04

## Tema: Sort y Listas Enlasadas

### INTRODUCCIÓN

aquí ira la intro

### MARCO CONCEPTUAL

aquí ira la MARCO CONCEPTUAL

### SOLUCIONES Y PRUEBAS

Ejercicio 1

//taylor

Ejercicio 2

Para lograr ejecutar el algoritmo de ordenamiento de insercion primero fue necesario crear la clase List



### LECCIONES APRENDIDAS Y CONCLUSIONES

aquí ira la LECCIONES APRENDIDAS Y CONCLUSIONES

### REFERENCIAS Y BIBLIOGRAFÍA

aquí ira la REFERENCIAS Y BIBLIOGRAFÍA

## 0.1. USAR COMO GUIA PARA EL INFORME

- Se tuvieron que implementar las funciones negative, join y under. En el codigo se detalla lo que hacen.

Listing 1: Picture version 1

```
1 import colors
2 from colors import *
3 class Picture:
4     def __init__(self, img):
```

```
5     self.img = img;
6     def __eq__(self, other):
7         return self.img == other.img
8     def _invColor(self, color):
9         if color not in inverter:
10             return color
11         return inverter[color]
12     def negative(self):
13         """ Devuelve un negativo de la imagen """
14         neg = []
15         for fila in self.img:
16             cadena = ""
17             for color in fila:
18                 cadena += self._invColor(color)
19             neg.append(cadena)
20         return Picture(neg)
21     def join(self, p):
22         """ Devuelve una nueva figura poniendo la figura del argumento
23             al lado derecho de la figura actual """
24         juntos = []
25         for i in range(len(self.img)):
26             juntos.append(self.img[i] + p.img[i])
27         return Picture(juntos)
28     def under(self, p):
29         """ Devuelve una nueva figura poniendo la figura p sobre la
30             figura actual """
31         return Picture(self.img[::] + p.img[::])
```

Para implementar el negativo se cambia el color de cada carácter, a excepción del espacio ya que este no tiene inverso con doble for(1 anidado). Juntando estos caracteres por medio en una cadena y despues juntando esta cadena al nuevo arreglo con el metodo append.

Listing 2: Ejercicio 2 a

```
1 from interpreter import draw
2 from chessPictures import *
3
4 draw(knight.join(knight.negative()).under(knight.negative().join(knight)))
```

Para poder hacer que se dibuje como se mostrara el la figura de abajo se tuvo de juntar un caballo blanco con uno negro en la primera fila; para ello se utilizo join y negative. Para la segunda fila se necesito saltar a la siguiente fila por lo que se utilizo el metodo under.



Figura 1: Ejecución exitosa ejercicio 2 a

## 0.2. Pregunta: Explique: ¿Para qué sirve el directorio pycache?

- Sirve para guardar los compilados de python, así como en java al ejecutarlos se crean los .class en python se crean los .pyc. Esto se hace automáticamente ya que se importan módulos de otras clases como se da en este caso.