

## 3 CABRISAS AMUEDO VIOU YIRMEIAH

### 3.1 Sobre el repositorio

Agrega el archivo `requirements.txt` donde se especifica qué bibliotecas de terceros son necesarias para correr tus programas y libretas. Puedes generar este archivo activando tu ambiente virtual y posteriormente corriendo el comando `pip freeze > requirements.txt`.

Bien en subir un archivo `LICENSE`, y mejor aún usando una licencia de software libre. ¿Puedes atribuirle una licencia a las funciones que les compartí para el gato? Si no les compartí una licencia, ¿es considerado como dominio público?

### 3.2 Sobre tarea 1

No veo tu tarea 1 y 2 en el repositorio, ¿no faltaron subirlas al GitHub? Recuerda que primero tenemos que agregar el código al *stage* usando el comando `git add`, luego empaquetar estos archivos agregados con un *commit* usando el comando `git commit` y finalmente, actualizar tu *origin* (el repositorio en los servidores de Bill Gates) usando el comando `git push`.

### 3.3 Sobre tarea 3

La función `is_final_state` no corresponde a un criterio para determinar si el estado `s` es final. Considera el siguiente estado:

```
s = [2, 1, 2, 0, 1, 0, 0, 1, 0]
```

Se consideran las posiciones `i` desde 0 hasta antes de la longitud de `s`, es decir, en el rango `[0, 8]`. En la primera iteración tenemos que `i == 0` y se verifica si `s[i-1] == 0`, es decir, si `s[-1] == 0`, es decir, si `0 == 0`. Ya que esto resulta en `True`, tu función regresa `False`, indicando que `s` no es un estado final, sin embargo si lo es, el jugador 1 ya ganó.

Buen uso de un diccionario `line_matrix` para almacenar los vecinos relevantes de cada casilla.

En `risk_calc` comentas que calcula el tamaño del riesgo de no jugar en determinada casilla, esta idea me parece excelente, sin embargo, la función regresa una lista, mientras que un tamaño es usualmente asociado a un entero no-negativo.

La implementación de `actual_winner` (y de `risk_calc`) es algo complicada, podemos rebotar algunas estrategias para implementar de forma menos complicada la idea que tienes de incorporar un *factor de riesgo* en la decisión de jugada por parte de la máquina.