

IADES - Institutos Argentino de Educación Secundaria y Superior

Carrera de Analista de Sistemas

Programación II - Profesor Pablo Scrigna / Taller de Programación II - Profesor Federico Polisenio

Examen Final en Grupo

Tema: Seguimiento del Puntaje de una Partida de Truco

Objetivo: Desarrollar un programa modular para hacer un seguimiento del puntaje de una partida de truco utilizando Python. La solución debe ser presentada a través de un repositorio en GitHub, donde cada integrante del grupo participe en el desarrollo y documentación del proyecto.

Instrucciones:

1. Formación de Grupos:

- El grupo estará conformado por los siguientes alumnos:
 - Gonzalez Gil Danielys
 - Lunda Ronaldo
 - Martinez Bayeto Sebastian
 - Milito Mario
 - Nava Juan
 - Ortiz Federico
 - Roldan Capponi Maximiliano
- Cada grupo deberá crear un repositorio en GitHub para gestionar el proyecto.

2. Estructura del Proyecto:

- El proyecto debe estar dividido en módulos según las funcionalidades descritas.
- Debe haber un archivo principal `main.py` que gestione la ejecución del programa.

3. Módulos y Funciones:

- **Módulo `flor.py`:**
 - `preguntar_flor()`: Función para preguntar si hubo flor en la ronda.
 - `gestionar_flor(puntaje_equipo_1, puntaje_equipo_2)`: Función para actualizar el puntaje en caso de que haya flor.
- **Módulo `envido.py`:**
 - `preguntar_envido()`: Función para preguntar el tipo de envido.
 - `puntos_envido(opcion, puntaje_ganador, puntaje_perdedor)`: Función para calcular los puntos del envido.
 - `gestionar_envido(puntaje_equipo_1, puntaje_equipo_2)`: Función para actualizar el puntaje en caso de que haya envido.
- **Módulo `truco.py`:**
 - `preguntar_truco()`: Función para preguntar el tipo de truco.

- `puntos_truco(opcion)`: Función para calcular los puntos del truco.
 - `gestionar_truco(puntaje_equipo_1, puntaje_equipo_2)`: Función para actualizar el puntaje en caso de que haya truco.
 - **Módulo `puntaje.py`:**
 - `verificar_victoria(puntaje)`: Función para verificar si un equipo ha ganado.
 - `mostrar_puntajes(puntaje_equipo_1, puntaje_equipo_2)`: Función para mostrar los puntajes actuales.
4. **Archivo Principal:**
- **`main.py`:**
 - Importar y utilizar las funciones de los módulos anteriores.
 - Gestionar el flujo principal del programa, incluyendo las rondas y la verificación de la victoria.
5. **Requisitos del Repositorio en GitHub:**
- El repositorio debe ser público y nombrado de manera identificable (e.g., `grupo1_truco_puntajes`).
 - Debe incluir un archivo `README.md` con la siguiente información:
 - Descripción del proyecto.
 - Instrucciones para ejecutar el programa.
 - Nombres de los integrantes del grupo y sus roles.
 - Estructura del proyecto.
 - Cada integrante del grupo debe hacer al menos un commit significativo, documentando su contribución.
6. **Entrega y Presentación:**
- La entrega del proyecto se hará enviando el enlace al repositorio de GitHub al correo del profesor antes de la fecha límite.
 - Cada grupo presentará su proyecto en Mesa de Final, explicando la estructura del código, las funcionalidades implementadas y cómo trabajaron en equipo utilizando GitHub.
 - Video de menos de 10 minutos mostrando la ejecución del programa y presentando al equipo de trabajo y sus roles en el trabajo.
7. **Criterios de Evaluación:**
- **Funcionalidad:** El programa debe funcionar correctamente y cumplir con los requisitos especificados.
 - **Modularidad:** El código debe estar bien organizado en módulos.
 - **Estilo de Código:** El código debe seguir las normas de PEP 8 para asegurar legibilidad y mantenibilidad.
 - **Colaboración:** Evidencia de trabajo colaborativo en GitHub.
 - **Documentación:** El repositorio debe estar bien documentado.
 - **Presentación:** Claridad y coherencia en la presentación del proyecto.

Fecha Límite:

- **Entrega del repositorio en GitHub y video en YouTube:** 30 de julio 19hs
- **Presentación en Final:** 1° de agosto 19hs

ANEXO

Puntaje

Flor

3pts

Envido

2pts

Real Envido

3pts

Envido Envido

4pts

Falta Envido

Si el rival está en las malas, el ganador gana el juego.

Si el rival está en las buenas, el ganador suma los puntos que le falta al rival para llegar a los 30pts y ganar la partida

Truco

2pts

Re-Truco

3pts

Vale Cuatro

4pts

Envido No querido (cualquiera)

1pt

Truco No querido

Truco: 1pts / Re-Truco: 2pts / Vale Cuatro: 3pts