CS_Dino

I want to reimplement the Dino Chrome (Yeah the existing one)

CS_Dino is a reimplementation of the classic Chrome Dino game using the `cmu_graphics` library. The base functionality will be inspired by the original endless runner, but the game will be extended with new mechanics, features, and levels to make it more engaging and challenging. These additions include combat elements, multiple levels, and possibly 3D elements if time allows.

The original Chrome Dino game is a simple and addictive endless runner with minimalistic pixel art, which is playable offline on Chrome.

My version will differentiate itself by evolving from a simple endless runner into a more dynamic, story-driven game with distinct levels. For instance, in later levels, players will face enemy dragons, manage health bars, and use fire attacks. If time permits, we will implement a final boss battle in a pseudo-3D environment. This creative extension of the Dino universe adds depth and replayability that sets our game apart from existing versions.

Algorithmic Plan:
The trickiest part of the project will be designing a smooth and fair **combat system** in Level 2 .

The plan:

- For enemy dragons, I will implement a basic finite state machine (FSM) to model enemy behavior: *Idle*, *Chasing*, *Attacking*, and *Retreating*.
- Combat logic will include hitboxes for fireballs, collision detection, and a cooldown system to prevent spam attacks.
- Health will be tracked for both player and enemy dragons, displayed as dynamic health bars.
- For Level 3 (later), a simplified 3D projection using 2D transformations will simulate depth and movement during the boss fight.

Overview

Existing features:

- Score
- Obstacles speed
- Day & Night cycles

Tentative new features:

- Levels
  Level 1: Just skipping and dodgin obstacles (birds & cactus)
  Level 2: Fighting with other dragons (health bars)
  Level 3: (If time allows), 3D with the big boss
- Enemies (chasing the dino)
- Spitting fire to defeat those enemies