This part is evaluated to assign a maximum of 2 extra-points for qualified students taking the exam with a mark >= 18

Pac-Man

Pac-Man, originally known as Puck Man in Japan, is a maze video game developed and released by Namco in 1980. The game was later released in North America by Midway Manufacturing. The inspiration for the Pac-Man character came from a pizza with a slice removed, and the game's characters were designed to be cute and colorful to appeal to younger players. In the game, Pac-Man eats the pills scattered throughout the maze to make points, while avoiding the ghosts that chase him. You can try online Pac-Man here.

1) **Implementation details for LandTiger Board**

In Keil µVision, use the LANDTIGER **board** (the emulator is not sufficient this time!) to implement a Pac-Man game.

Please deliver a zip folder with all the files of your project (you must save the project with all the compilation options you used).

Example: extrapoint2.zip

Specifications

You must **extend** the Pac-Man game you have created for Extrapoint 1, adding new functionalities, meaning that **the project for Extrapoint 2 must respect the same specifications listed in Extrapoint 1, plus the new ones**.

Spec. 1) Create an AI-controlled **ghost** that pursue Pac-Man, causing him to lose a life in case of contact.

The ghost has a unique AI (artificial intelligence) and movement style:

1. **Blinky (Red):**

   o Blinky directly follows Pac-Man, always trying to reach him. As the game progresses, he becomes faster and more aggressive.

The ghost in Pac-Man (Blinky) is the main antagonists of the game. His primary goal is to chase Pac-Man and "eat" him. However, his behavior changes based on the game mode and his individual strategies. Implement two strategies:

1. **Chase Mode:**
   o In this mode, the ghost actively tries to catch Pac-Man.
2. **Frightened Mode:**
   o This mode activates when Pac-Man eats a "Power Pills". The ghost turns **blue** for **10 seconds**, runs away from Pac-Man, and can be eaten by Pac-Man for extra 100 points. If the ghost is eaten, it will respawn after 3 seconds in its central spawning point (the empty square that is also the initial starting point).

Hint: the ghost's AI can be implemented by using an appropriate pathfinding algorithm, e.g the A* algorithm, that finds the shortest path between the ghost's current position and his target. The differences in behavior may be implemented by selecting the appropriate target (and changing it during the game if needed)

Spec. 2) Configure the **speaker** to play sound effects and background music using the speaker.

Spec. 3) Use the CAN peripheral to print the current **Score** of the game, the **Remaining Lives** and the **Countdown timer** through the **CAN bus**, as Figure below shows.

You must adequately configure the CAN controllers of the LandTiger Boards. It must be configured to be in what is called external "loopback" mode. This means that the board will communicate the message back to itself! To achieve this goal, you will use CAN1 and CAN2, one set to receive the message (CAN2) and one set to send the message (CAN1). Since these are two different channels, they don't actually know that they are communicating with the same board! So by all means, you are communicating through CAN with another board. The message that you have to send should be encoded in the following way:

Remaining time	Remaining lives	Score
8 bits	8 bits	16 bits

Every **message** must be saved in a **32-bit unsigned int variable**