Purpose:

We are making an alarm clock that prompts users to perform a series of challenges in the morning to prevent them from falling back to sleep upon waking up:

- the sound of the alarm clock will not stop until the user has completed his/her challenges
- during the challenge, the alarm clock will temporarily snooze until the user either completes the challenges, after which it will permanently stop, or until the user fails, after which the alarm will sound again
- types of challenges include ones that encourage the user to engage in physical activity, such as a game where the user has to shake their phone to stop the alarm clock, and ones that make the user think, such as math problems.

Software Components:

- Implement a LCD interface that allows the user to interact with the game
- Implement a LCD interface that allows the user to set the alarm clock, adjust the time, and keep track of time

Prototype Plan:

Each minigame that prompts the user to feel more awake is its own separate widget. Therefore, to start, we can treat the programming of each minigame as its own separate project. We will initially create several experimental prototypes before stringing them together to complete the alarm clock. Additionally, we may run into challenges while programming a minigame that may force us to pivot. Thus, our evolutionary prototype will involve us either making changes to our minigame to make them easier to program or switching minigames altogether.

Hardware:

- Arduino Board
- Arduino Sound Speaker
- LCD Display
- Buttons
- Wires
- Accelerometer

Challenges:

- Figuring out how to use the Arduino board and IDE
- Displaying numbers and complex images on a LCD Display
- Creating a dynamic game on a LCD display