

# CHECKERS

Dang Nguyen, Oscar Wang, and Mark Wu, Winter 2018

## OVERVIEW

We propose to create a 2-player checkers game that takes inputs of the column and the rows you want to move a checker in through buttons, and then visualizes the gameplay with a bunch of LEDs.

## PERIPHERALS

1. 8x8 LED matrix <https://www.adafruit.com/product/902>
2. Buttons

## SOFTWARE DESIGN

We will use a single while loop that takes in a function that waits for a GPIO interrupt from the button, and then check if the current move is legal, and if it wins the game for the current player. Each player has their own color for the LED that represents their checker; note: when a checker makes it way to the end the opponent's side of the map, it is crowned and becomes a king; to implement that the LED will have a distinct color. Instructions, for example current player and winning messages, will be sent to termite through UART.

## GOALS

The goal of this project is to take in GPIO interrupts to control LED displays with the logic of CHECKERS in the backend.

1. Pieces can only move diagonally and cannot move backwards unless they have been 'crowned'
2. To be 'crowned', the piece must reach the end of the opponent's side of the board
3. Crowned pieces can move forward and backward
4. First player to eliminate all opposing pieces is the winner

## GROUP RESPONSIBILITIES

Dang Nguyen - in charge of algorithm and implementation for checker movement animation, and debugging

Oscar Wang - in charge of algorithm and implementation to win the game

Mark Wu - in charge of algorithm and implementation to detect GPIO for checker moving functionality