

## A simple Snake game

By Malcolm Liljedahl (940912-1473) and Harry Lazaridis (000520-0076)  
2021-03-06

**Objective and Requirements.** Our goal was to make a snake game in C and being able to display the game on an ordinary VGA-monitor. But we were not able to achieve our goal, due to bad planning and lack of time.

Though we made a simplified version of this project, which is described below. The game has a "snake" that is controlled by the user via the I/O shield buttons. The goal of the game is the "eat" fruits and gain as many points as possible, you get 1 point per eaten fruit. When a fruit is eaten the tail of the snake is extended. If the snake touches the walls around the game area, the game is instantly over.

**Solution.** The game is developed using only the supplied ChipKit Uno32 together with the I/O shield. The game is being played on the OLED display and the controls are being made via the push buttons. The points are shown as binary numbers via the LEDs. A problem was to obtain a random spawn timer for the fruit, we managed to do this by looking at another project in this course from a previous year and then we could implement a variant of it in our own program.

**Verification.** First we looked through the code that was given to us via the labs, to see what we could use. We built the base of our program using the lab3 programs, to initialize the OLED display, handle the timing etcetera. Then we started by building our game field and then we tried implementing each element one and one, so we could try every part of the game. This meant that we could debug the code after each step and not only when we were done.

**Contributions.** The plan was that Harry was going to be the sole game logic developer and Malcolm was going to handle the i2c, I/O and the VGA connection, but since we only had time to make the game on the pic32, we did it all together. We took turns writing the code while the other person was watching and constantly discussing how and what to write etc.

**Reflections.** Most of the game works as we intended. We learned that when entering a new area (hardware programming) you should always start early, instead of pushing it towards the last week as we did.