Hi Matt.

Meant to send this to you earlier. I have some ideas we could use for our videogame.

First, we can have structs for enemy and tower. They might look a little like this:

struct enemy {

short x;

short y;

short strength;

short alive; //1 or 0

}

struct tower {

short x;

short y;

short rate; //rate of firing

}

We can also have a struct for the board layout:

struct board {

char[10] file\_name; //name of bmp image that holds the

//picture of the path to be followed

path\_corner[10];

}

The struct path\_corner would indicate where (x,y) on the board is a corner of the enemy’s path. When an enemy crosses a path\_corner, the path\_corner would change the enemy’s direction. In this way, we can delimit the path using less memory than if we had some list with every single pixel the enemies need to follow.

struct path\_corner {

short x;

short y;

direction; //(“north”, “south”, “east”, “west”)

}

The basic ordering for our videogame might look like this:

1. Load main menu.

2. Use nunchuk/usr\_button to press play.

(OPTIONAL) Main menu has buttons for play and an about button which would load a screen that tells the user about the game and about us. Would be easy and adds a feature technically to the game

3. Board appears.

4. Player can place towers using cursor guided by nunchuk joystick.

5. Player presses button (c, z?) to play.

Use c button to select things (like left click on a mouse) to select towers to place. Use Z to play or pause the game

6. Enemies begin appearing on path.

7. Towers shoot at closest enemy to finish.

8. If an enemy reaches finish, game over and pop up a botton for retry the round or game