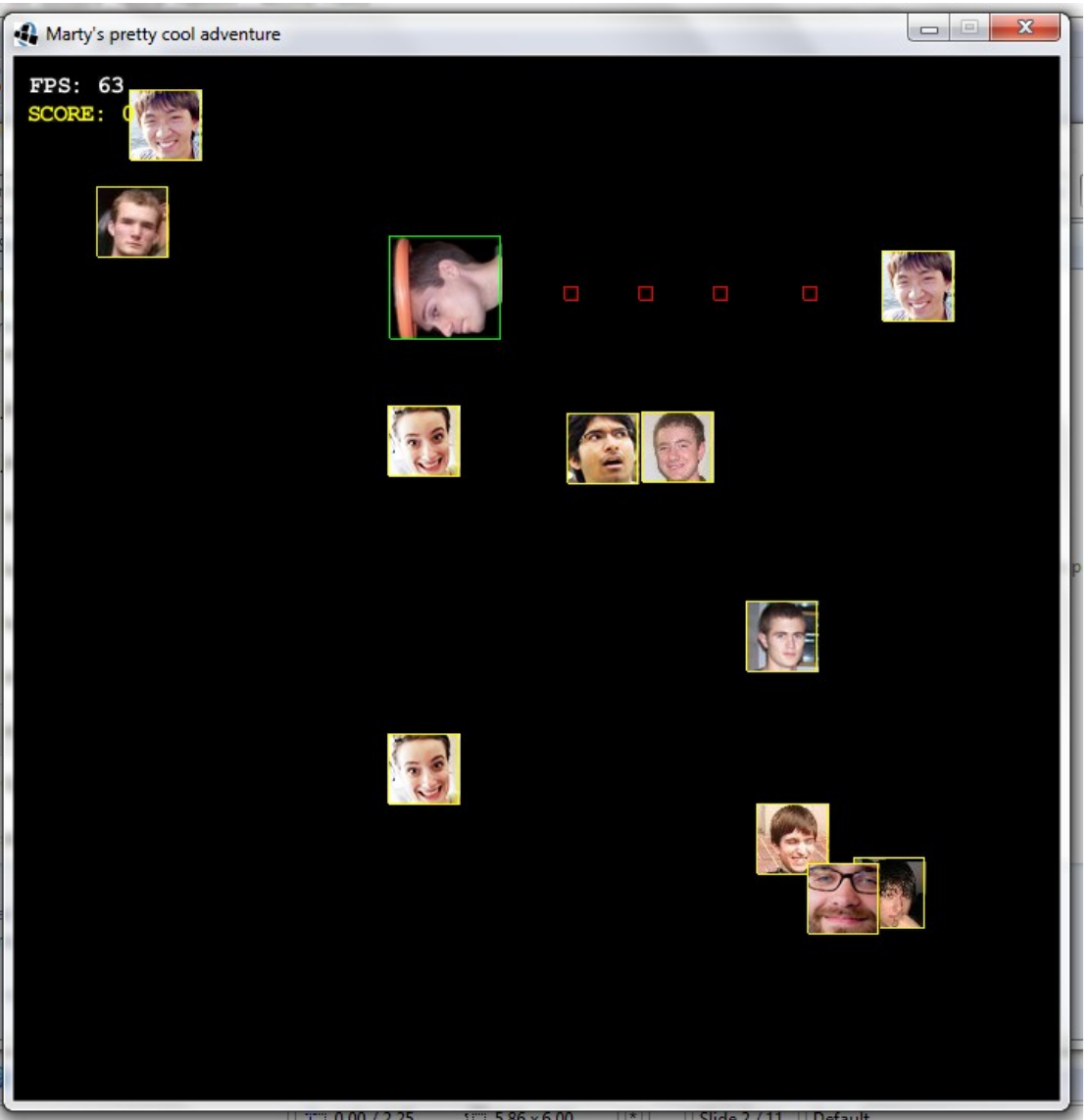
Flash {AS}3

Session 3: Events

Shiny Yang (mootothemax@gmail.com), special thanks to Alex Miller

Flash Homework

Recreate this game
(in flash)



Playable online at:
http://www.spotcos.com/martygame.html

Flash Homework

What do we need to know?

How to draw player + enemies on screen?

Add sprites to the display tree. (We already know this)

How to handle keyboard input?

Set up event listeners for keyboard events.

How to make an update cycle?

Set up a timer event.

How to detect collision?

Use the sprite's hittestobject() method.

Syntax Review

```
Sprite.addEventListener(MouseEvent.CLICK, function(e:MouseEvent){});
```

Syntax for creating “click” event listener.

Works with almost every DisplayObject
(and more importantly, every sprite)

Compared with javascript (Note: NOT the prototype .observe() method):

The function will have local scope based on where the function is defined. (Compared to non object-oriented model for js)

this keyword in the event function will NOT refer to the “called” object (unlike in javascript)

optional event parameter will need to be of type of the event (see e:MouseEvent)

ActionScript Syntax

Keyboard Events

```
stage.addEventListener(KeyboardEvent.KEY_DOWN, keyTest);
 stage.addEventListener(KeyboardEvent.KEY_UP, keyTest);


function keyTest(e:KeyboardEvent) {
    trace(e.keyCode);
}
```

KEY_DOWN is called semi-continously when a key is pressed

KEY_UP is called when a key is released

To find out which key was pressed, access the event parameter's .keyCode parameter, then compare to:

```
Keyboard.SPACE
Keyboard.UP
Keyboard.DOWN
Keyboard.Q
```

(And so on..or just remember the keycodes themselves)

ActionScript Syntax

Notes About Keyboard Events

Keyboard events should technically always be added to the stage.
(If they are added to a child, they will not register unless the child is focused)

`KEY_DOWN` does not fire in a consistent manner after the inital call

`KEY_DOWN` stops firing for a first key after a second key is pressed, and does not re-fire if the second key is released but the first remains pressed down

(ex. hold down 1, then hold down 2, then release 2, does not fire 1 events)

A problem for many games/applications where precision control is required. Fixable through a combination of `KEY_DOWN` and `KEY_UP` (see Vidya.as keypress methods for my “input stack”)

ActionScript Syntax

Timer Events

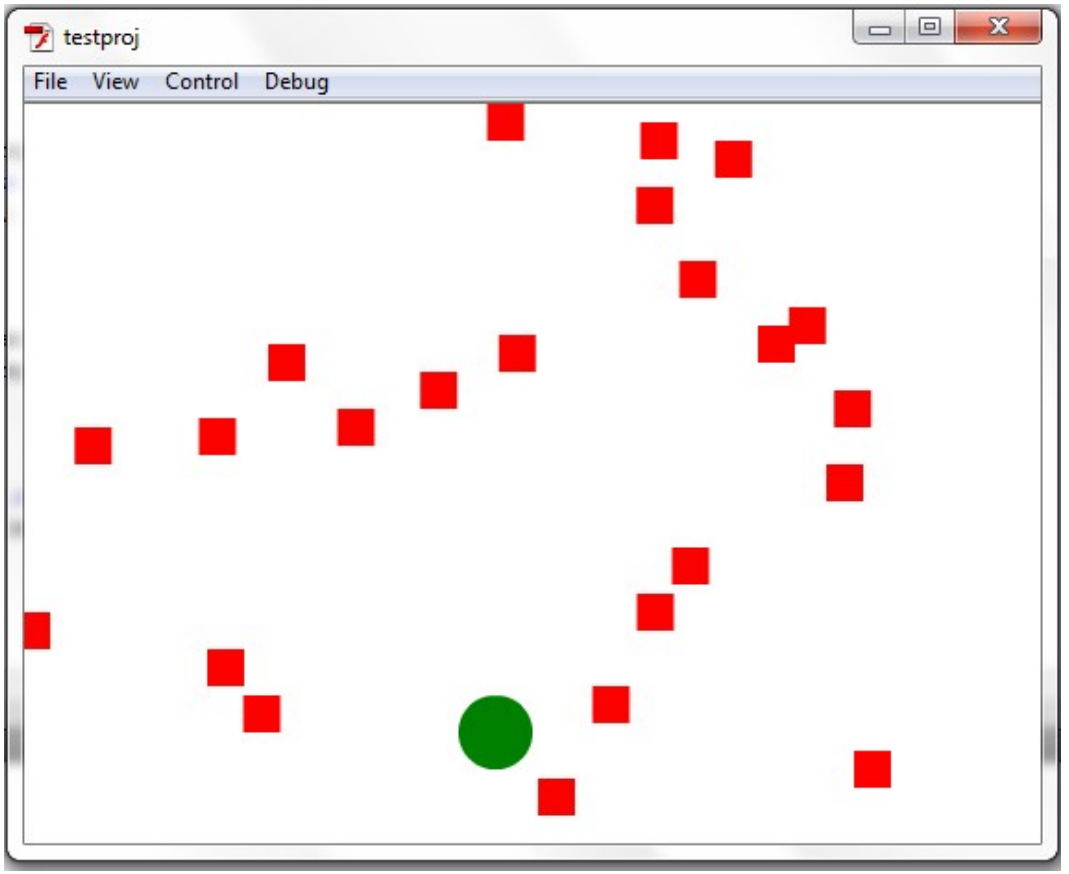
```
//first param is MS delay, second is repeat count (none is infinite)
var timer = new Timer(30);

timer.addEventListener(TimerEvent.TIMER, function(){});

timer.start();
timer.stop();
```

Once started, the timer calls the event function once every delay ms for repeat times.

The Timer variable should be accessible as a variable/field if you want to stop/edit the actions.

ActionScript Example

BlocDodger Flash



1. Have a player (green ball) that is movable left and right.

2. Have an update cycle that spawns red boxes and makes them fall 1px down.

(HINT: have a enemyarray array, also a method for creating boxes at random place)

also syntax note:
for each(var enemy:Enemy in ...) {

ActionScript Syntax

Hit Detection

```
Sprite.hitTestObject(obj:DisplayObject);
//returns boolean
```

Checks if the Sprite's graphics (or any of its children's graphics) intersect the graphics of another DisplayObject (or any of its children).

Pixel based collision detection that comes as default functionality for every Sprite. (Sure beats implementing your own hitboxes)

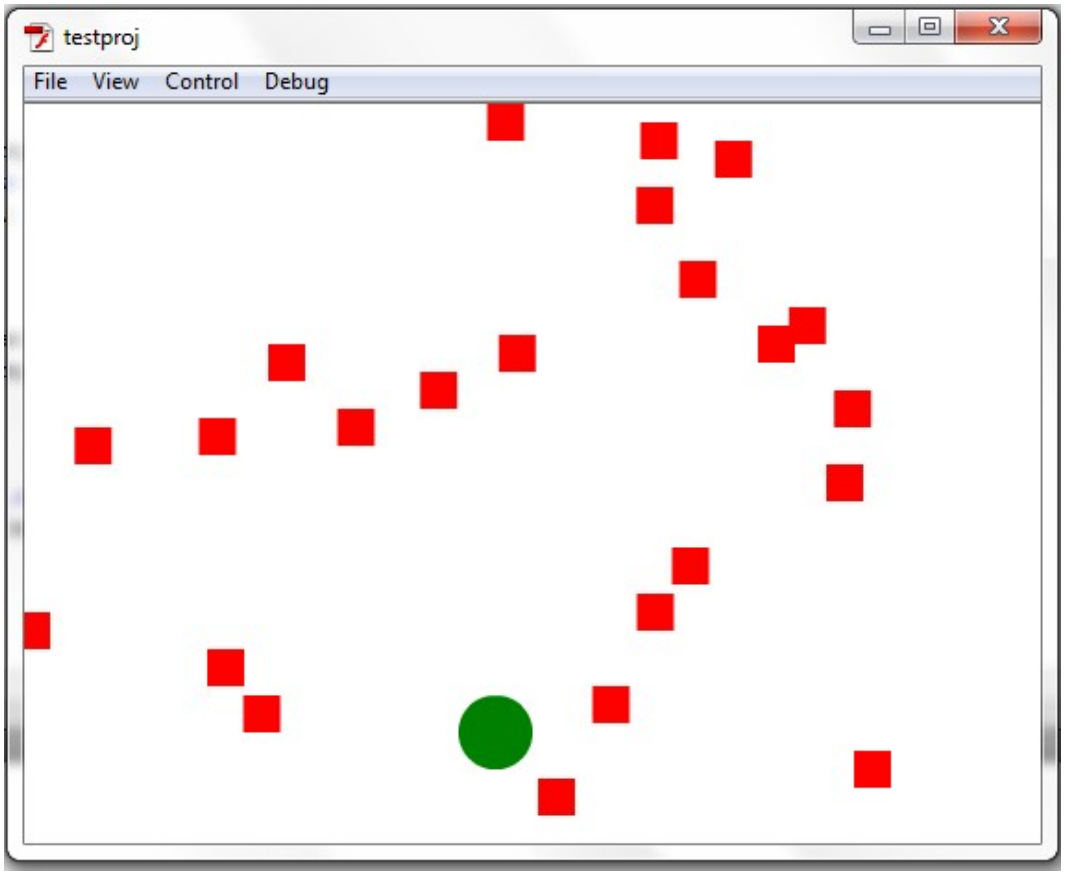
ActionScript Example

BlocDodger Flash

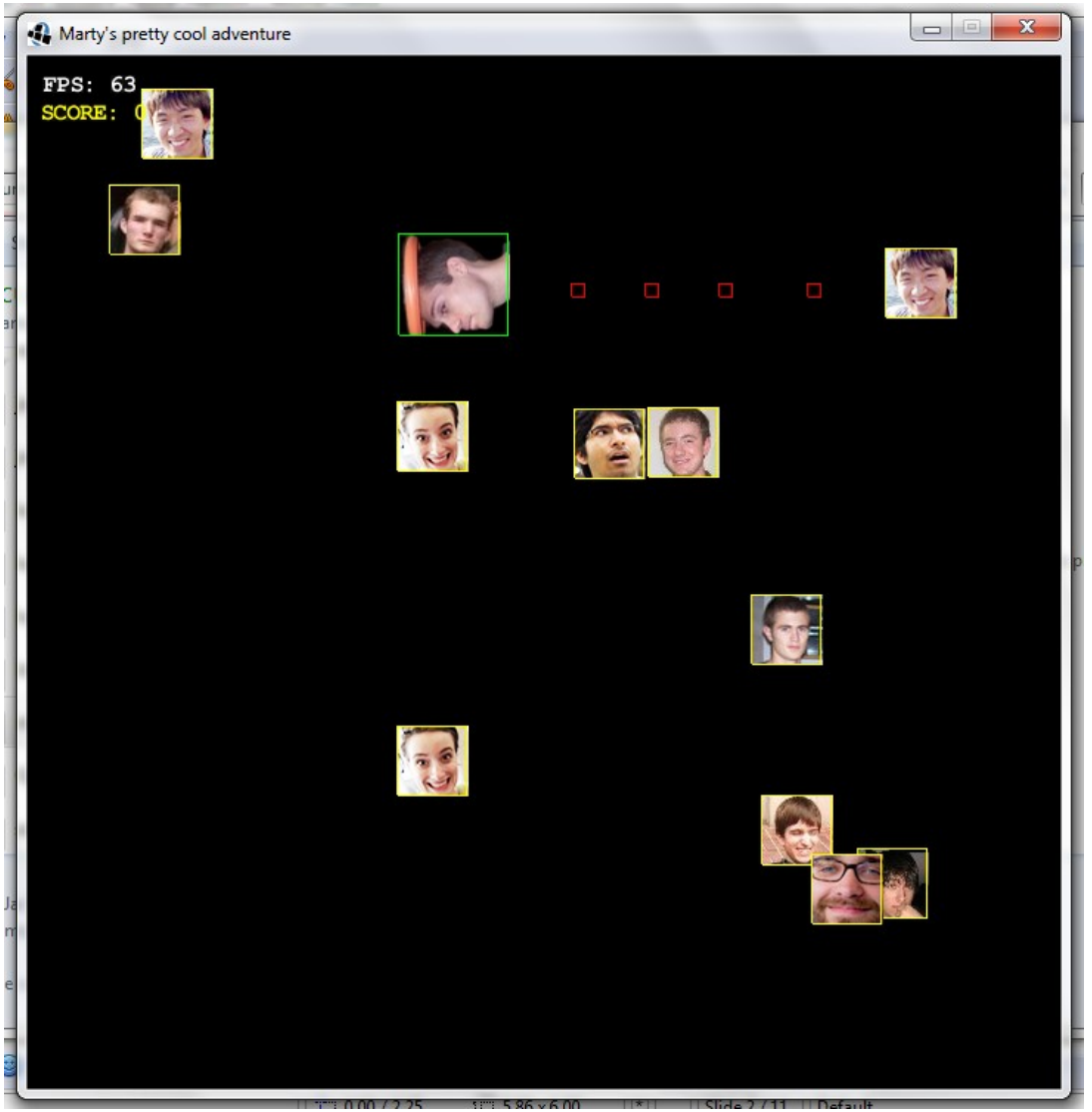
3. Have the game stop whenever an “Enemy” intersects the “Player”

4. This program is leaking memory. Can you figure out why and how to fix it?

syntax note:
array.splice(indexOf(pointer),1);



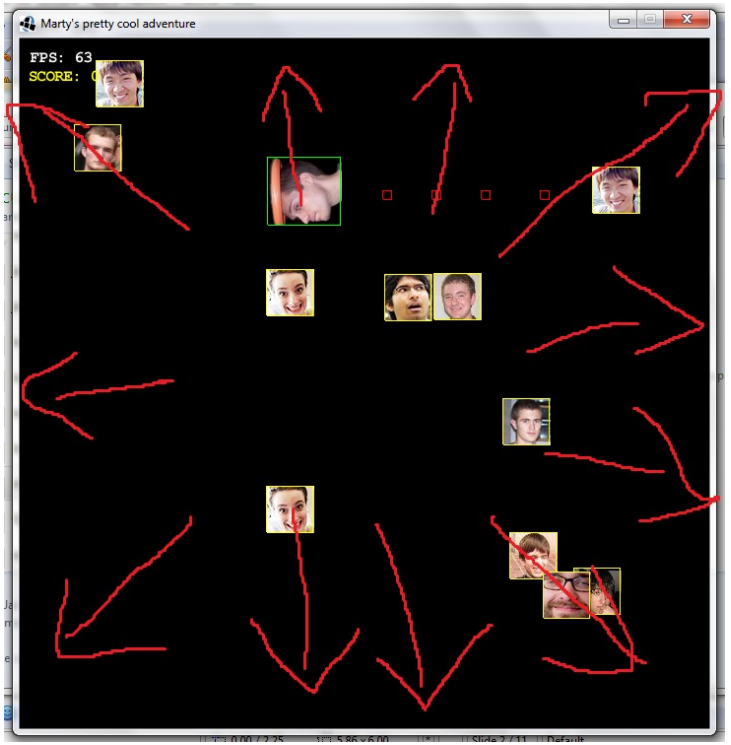
Flash Homework



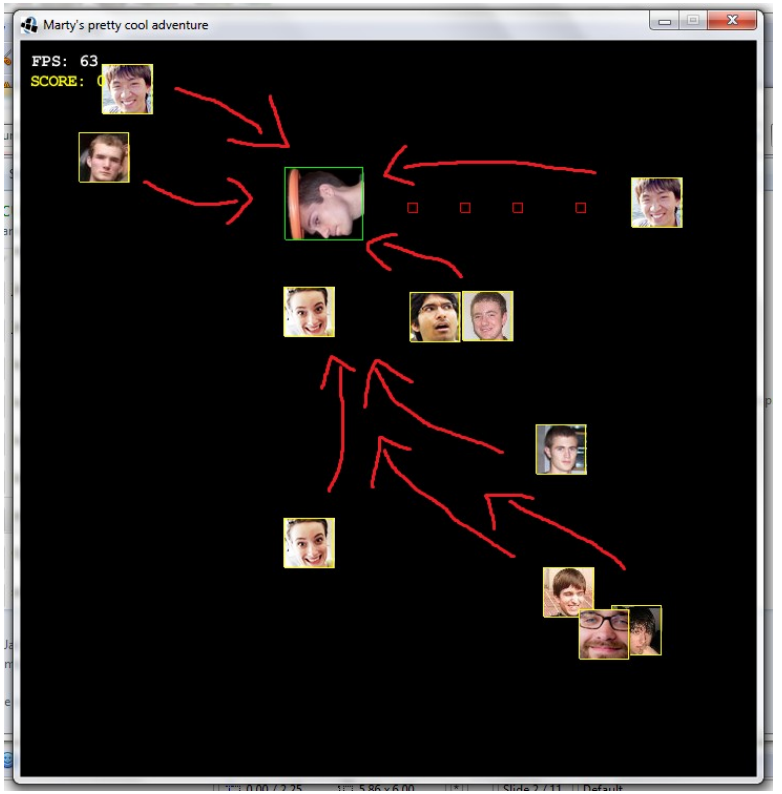
Recreate Marty's Pretty Cool Adventure! (without the marty)

Playable online at
http://www.spotcos.com/martygame.html

Flash Homework



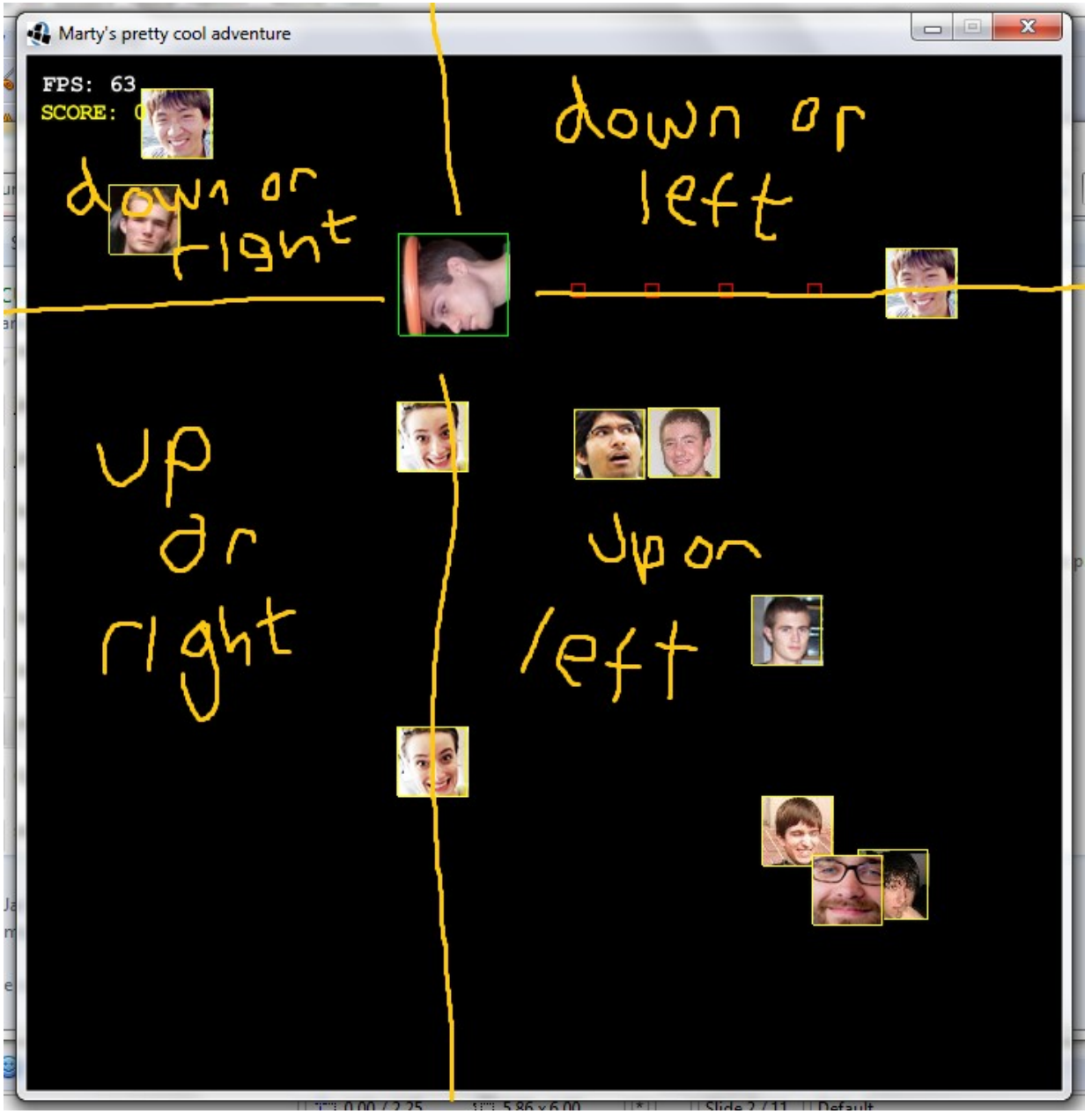
Enemies should be able to spawn from any of the 4 edges.



Enemies should always be moving towards Marty.

(Hint: they should all be in an array that is looped through every update cycle)

Flash Homework



The tracking algorithm:

Every update cycle, figure out which quadrant you are (in respect to the player).

Move one of the two possible directions given your quadrant.

(Can you improve it?)

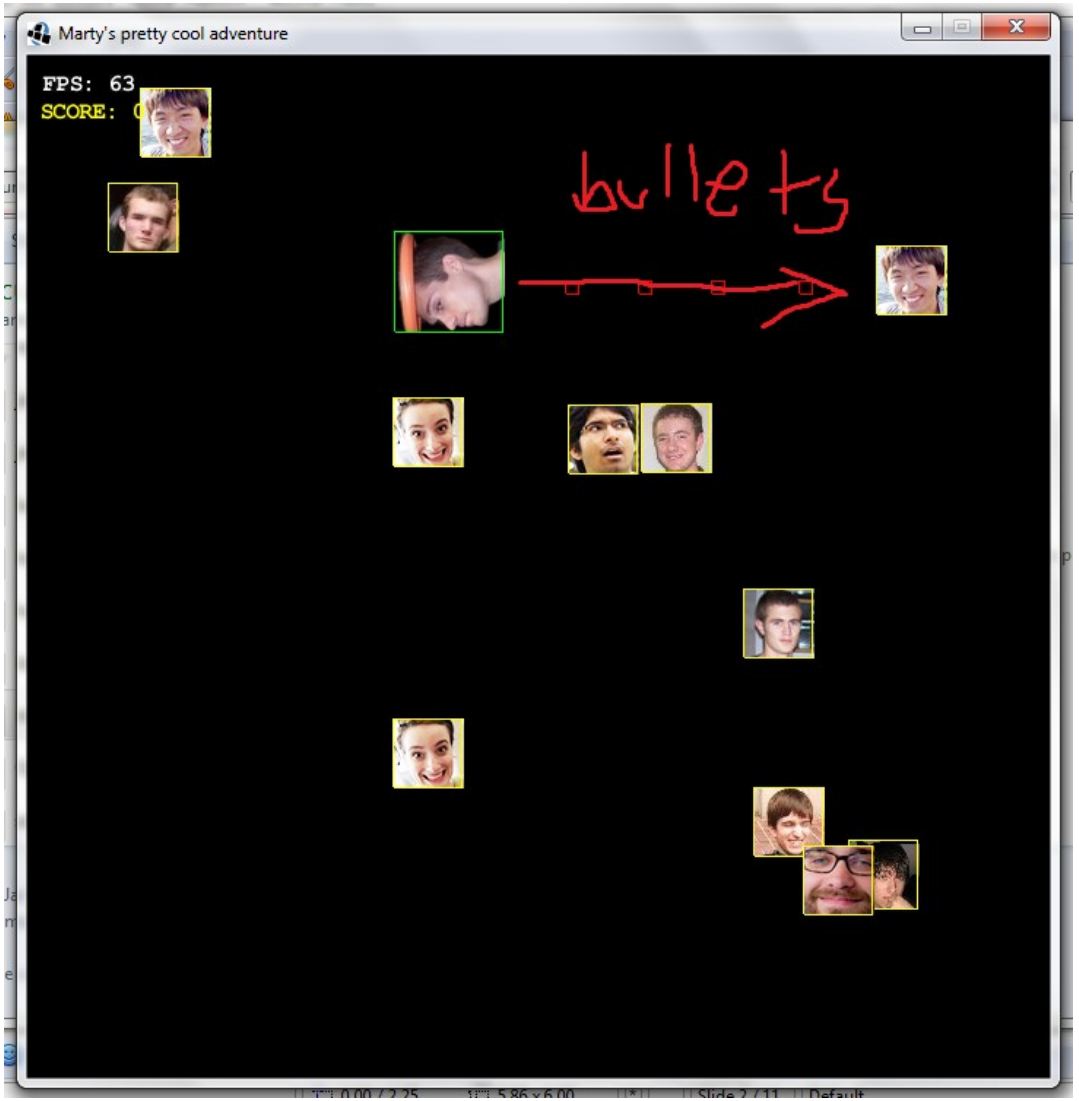
Flash Homework

Game Behavior:

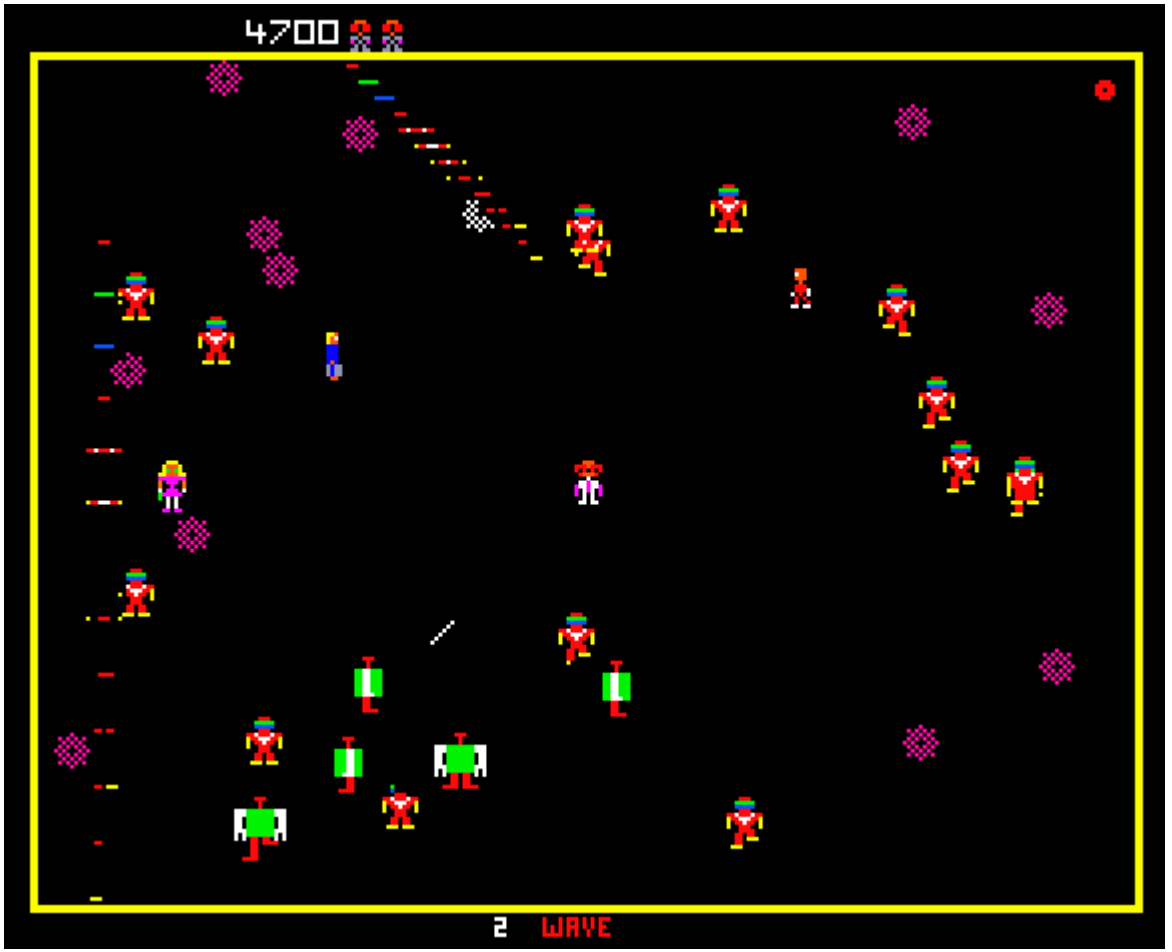
If one of the enemies touch the player, the game should be “over” (some sort of state change should happen)

Bullets should be fired FROM the player's location, based on the LAST direction the player moved.
(Bullets should be in an array of their own, also probably NOT a child of the player)

If a bullet touches an enemy, the enemy should be removed.



Flash Homework



Finally, add in at least one feature of your own.

Some ideas:
Another type of enemy?
Lives/points/menu?
Powerups?
Multiplayer?

And above all
HAVE FUN