Shane Smith
1112384
Dr. Mohammed
COMP4478

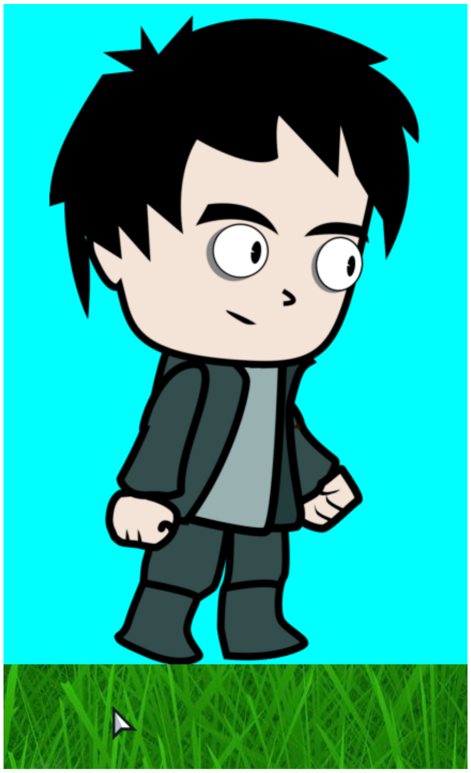
EXERCISE 1 - Animate a player in haxeflixel

SPRITESHEET
Created a smaller version (scaled to half the size) of the sprite sheet provided using each of the 6 walking animations and the TexturePacker program.

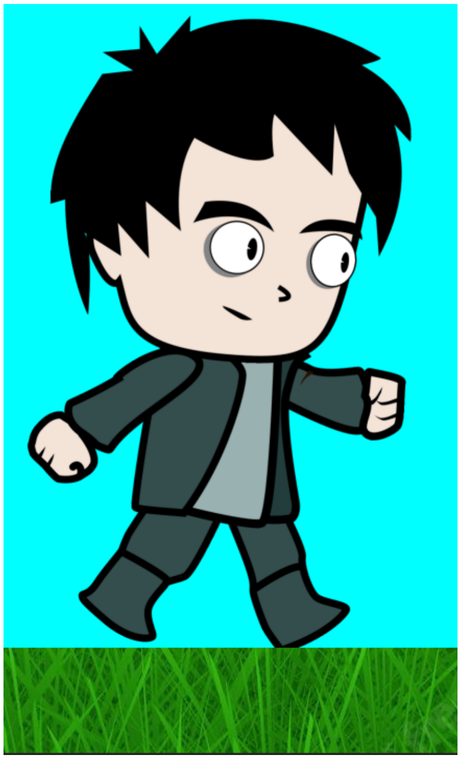
STRATEGIES FOR ANIMATION
In order to animate the sprite, I first created a player class under Player.hx which extends the FlxSprite class. This class will be used to animate the player. The player has a variable 'SPEED', which is used to determine the player's velocity.
I overrode the new() function in order to instantiate the player at the correct position, and to add the spritesheet as a graphic for the player. Each sub-sprite of the spritesheet is 384x632. I set the drag of the player in order to aid with the animation. I then created the "walk" animation, which uses all 6 images of the sprite sheet and moves at 12 frames per second. I also added an additional animation "idle" which uses the 6th and final image in the sprite sheet only, which is to be used when the player is idle. Finally, I set a facing flip for both left and right to determine what way the player will be looking when moving in either direction.
I added a new function called movement(). Here, we set the variables 'left', which is encompassed by pressing either the A or the left arrow key, and the variable 'right' which is encompassed by pressing D or the right arrow key. I then added some conditional logic to determine which animation will be playing. If the player is pressing one of the keys encompassed by either the left or right variables, the walking animation will play. If not, the idle animation will play. Then, more logic is added in order to determine velocity. If one of the left keys are being pressed, the player will face left and move left at -SPEED velocity. If one of the right keys is being pressed, the player will face right and move right at SPEED velocity. If both one of the left and one of the right keys are being pressed simultaneously, the walking animation will still be played, but the player's velocity will be set to 0. Finally, I added some additional logic to ensure the player does not walk off the screen.
I overrode the update() function to simply call movement().
In PlayState.hx, the background color was set to cyan and the player was instantiated and added to the state. I then added a ground sprite in order to look like the player was walking on something and not just in thin air. Finally, I added some informational text to the top left to describe how to move the player.

SCREENSHOTS OF ANIMATION

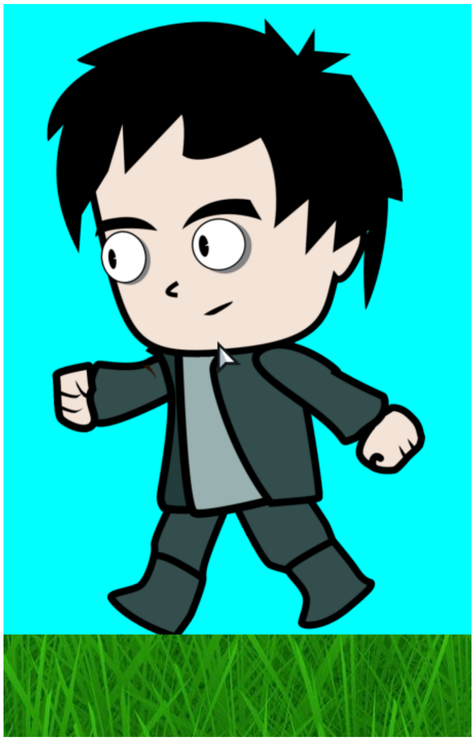
Idle animation



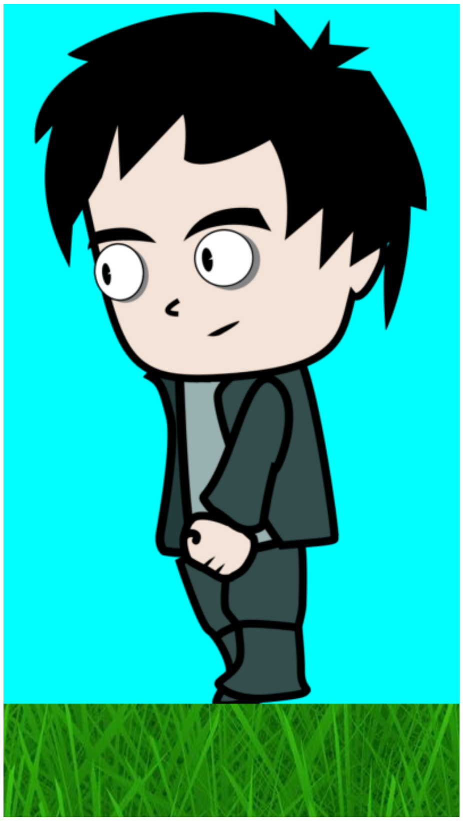
Pressing Left and Right Simultaneously



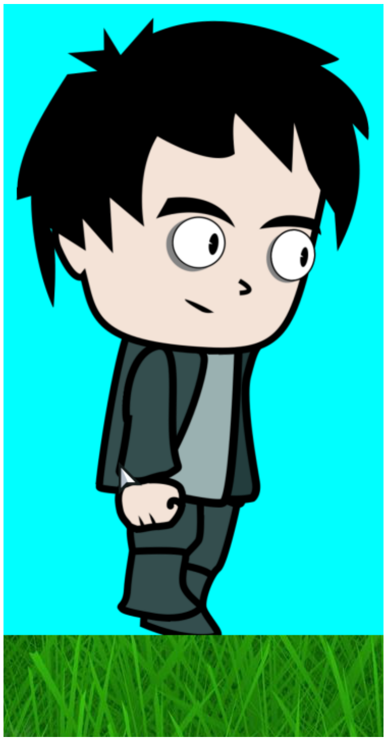
Walking Left 1



Walking Left 2



Walking Right 1



Walking Right 2