Lab 1: Little Fish, Big Fish

January 23-25, 2017

Welcome to CS 105 Lab Sections!

As part of your CS 105 experience, each week you will complete a **lab activity** that reinforces the concepts learned from your weekly activity and from lecture. This week, you learned that computer programming requires the development of an **algorithm** – a step-by-step procedure for the computer to run. These algorithms are then expressed in a **computer programming language**, which provides instructions to the computer to **run**, or "**execute**", your algorithm.

In both the weekly activity and lecture, you have been working with a visual, block-based computer programming language called Scratch. Today, you will write your **very first program**! Don't worry: it will be very simply, this sheet will guide you through it, and you have both TAs and peers to help you out!

Getting Started

Each week, you will work with other students. Each team will be 2-3 students. Once you form a team, log into your machine and, while your computer is logging in, **introduce yourself to your teammates** and find out:

- What are their names?
- What are their majors?
- What was the most interesting they did over the summer?

Once you are logged in, open a web browser and go to the CS 105 website: **http://courses.engr.illinois.edu/cs105**. Navigate to the **"Schedule"** page and click **"Lab 1"**. Follow the directions to be able to edit the bigfish project.

Little Fish, Big Fish

Once you have loaded the Scratch project file, you will find that the Scratch environment has four sprites already on the stage: three smaller fish around the edges and one larger fish in the middle.

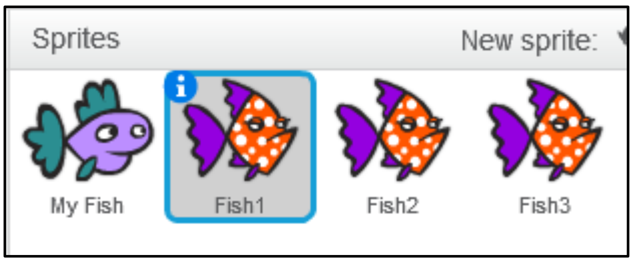
During this week's lab, **you and each member of your team** will modify their own copy of the Scratch project in order to program a simple game in Scratch. In doing so, you will:

- Program each of the smaller fish to swim across the screen,
- Program the larger fish to move in response to a user pressing arrow keys, and
- ***+1 Problem***: Program the smaller fish to disappear when eaten by the larger fish.

> **Before continuing,** spend a couple of minutes talking to your team about how you might complete this with what you already know about Scratch. Feel free to try it out inside Scratch before reading how to do it on the next page – experimentation helps learning!

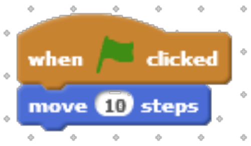
Program 1: Swimming Small Fish

Before starting on the program, you need to make sure you are programming the correct sprite. On the bottom-right side of the screen, you should find the collection of four sprites: "My Fish", "Fish 1", "Fish 2", and "Fish 3". **Click on the "Fish 1" sprite to select the "Fish 1" sprite.**



In order to create the effect of a fish swimming, we need the "Fish 1" sprite to **move across the screen forever**. Since we have the "Fish 1" sprite selected, any blocks we add to the sprite area will control only "Fish 1". To complete this:

- In the "Motion" category of blocks (blue), drag the move 10 steps block into the coding region.
- In order have Scratch run the command, we need to attach an event to the move action. In the "Events" category, drag the when ⚑ clicked block and connect it to the move 10 steps .



```
when ⚑ clicked
move 10 steps
```

Click the green flag: ⚑. At this point, each time you press the flag the fish jumps forward a little bit! For possibly the first time, you have programmed a computer – congratulations! However, we won't stop here. We need to complete the fish swimming across the screen. To do this, we need the program to run forever:

- In the "Control" category, drag the **[forever […]]** loop out.
- Rearrange the blocks so that the **move block** is inside the **forever loop** and the **forever loop** is connected to the **when flag clicked** staring block.

The lab continues on the reverse side of this sheet of paper.

Click the green flag: ⚑. Now, the fish should fly across the screen and end up with its head buried into the edge of the screen! Let us help our poor fish out by telling him to bounce off the edge:

- In the "Motion" category, find the **[if on edge, bounce]** block near the bottom of the list.
- Place the **if on edge block** inside the **forever loop** after **the move block**.
- While were here, lets also slow down the fish. **Change move [10] steps to move [2] steps** by editing the value inside the text box inside the **move block.**

Click the green flag again: ⚑. Now that looks like a swimming fish!

- Now **re-program this program again for "Fish 2" and "Fish 3" sprites** by clicking on them in the sprite area found in the bottom-left area of Scratch. All three smaller fish should move across the screen when you press the flag button to start it!
- Finally, **make each of the three fish move at a different speed** by having each program have a different number of steps that a fish moves at a time. *(Keep the numbers less than 5 steps.)*

Before continuing, make sure that you and all your teammates have all three of your fish moving across the screen at different speeds, bouncing off the walls when they reach the walls. Make sure you and all your team members understood this page.

Program 2: Moving the Larger Fish

Now that we have taken care of the smaller fish, this program will focus on the larger fish found in the center of the screen. As with the smaller fish, **click on the "My Fish"** sprite to begin programming the larger fish. This will cause the programming area to go blank since we have not programmed the "My Fish" sprite yet.

For the large fish, we want to have our arrow keys control the movement instead of it automatically moving across the screen. To do this, we require checks -- called **conditionals, represented as diamond-shaped blocks** in Scratch – to see if the user is pressing an arrow key.

- Since we also want this to run forever once we click the flag, begin by placing a **[when [flag] clicked] block** *("Event" category)* and a **[forever […]] loop** *("Control" category)*.
- Inside the **forever loop**, we need to check if the user is pressing the up button to move the fish forward. Inside the "Control" category of blocks, find the **[if <> then […]] block** and place it inside of your **forever loop**.
- Inside the diamond-shaped conditional region of the **if block**, place the **<key [space] pressed>** conditional block from the "Sensing" category. Once the **key pressed conditional** is inside the **if block**, change the selection from [space] to **[up arrow]**.
- At this point, the **if block** reads: **"if <key [up arrow] pressed?> then"**. Now we simply need to program the computer to move forward when the up arrow is pressed. To finish this, simply drag a **[move [10] steps]** block form the "Motion" category to inside the **if block**.

Click the green flag: ⚑. The three smaller fish should still be moving back and forth across the screen (from the first part of this lab). But now, when you press the up arrow on your keyboard, the larger fish should now move forward! ***This is awesome***, except our poor fish will starve if he can't navigate to the other fish. ☹

To allow the fish to eat, program the fish to turn when you press the right arrow keys:

- Add another **if block** into your program, placing it outside if your **existing if block** but inside your **forever loop**.
- Add the **<if [right arrow] pressed?>** conditional to your new **if block**.
- Inside of your new if block, find the **[turn [right] [15] degrees]** block in the "Motion" blocks.

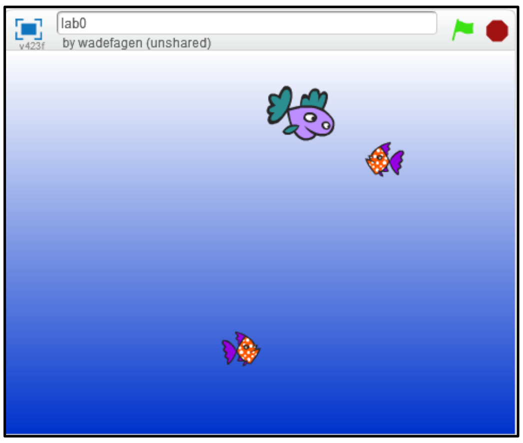
Click the green flag: ⚑. You should now be able to use the right arrow key to turn the fish and the forward button to make him move forward. Let's finish this up:

- Change the **turning block** to have the fish turn only by **[5] degrees** at a time.
- Add a new **if block** and **conditional** to program the fish to turn left when the left arrow key is pressed, just as you did with the right arrow key.
- Add a final **if block** and **conditional** to program the fish to move backwards when the down arrow key is pressed. To do this, the fish to should be programmed to **move a negative number of steps** when the down arrow key is pressed.

Click the green flag: ⚑. Your fish should feel very natural to move with the arrow keys. Feel free to adjust how much the fish turns per cycle or how much it moves. After all, it is your program!

Before continuing, make sure that you and all your teammates have all three of your fish moving across the screen and everyone can move their big fish. **This will earn you 5/5 points for this lab!**

Program 3: Little Fish, Big Fish



If you have the three smaller fish moving back and forth across the screen and the larger fish moving in response to the arrow keys, you have completed the required portion of the lab and will earn 5/5 points! Make sure to submit your program to the course website (see the next section on this page).

However, **if you still have time**, you can earn a 6/5 for this lab through the first ***+1 problem*** in CS 105! The description of what you need to do is quite simple:

- When the larger fish touches a smaller fish, the larger fish should "eat" the smaller fish. When this happens, the smaller fish should disappear (be hidden) from the screen.
- When a user presses the green flag, all fish should re-appear (be shown) if they were hidden.

To complete this, you will need to look through the list of possible blocks and find one to let you determine ***if*** the two sprites are touching in order to complete the program. Once you understand the

The lab continues on the reverse side of this sheet of paper.

algorithm that is required to determine ***if*** the two sprites are touching, you will need to think about **which sprite is being changed** and write your code within that sprite.

Since this part is extra credit, your TA won't be able to help you in lab on this part. However, you should continue to work with your team to figure it out!

This part is designed to be more difficult and you should not worry if you are unable to get this part. There will be numerous other chances to grab +1s!

Submission and Grading

Once you have finished or when the lab time is over, submit your program to the "Lab 1" page on the CS 105 website. Even if you weren't able to complete everything, make sure to submit your lab for some credit. **In order to submit a Scratch project, you must download the Scratch project to your computer. To do this:**

- Inside Scratch, select the "File" menu and click "Download to your computer"
- In the dialog that opens up, navigate to your desktop
- Save the file as **`lab1.sb2`** to your desktop

In a new tab in your browser (making sure to leave Scratch open), go to the CS 105 website and navigate to "Labs", "Lab 1", and to the submission page. On the submission page, upload your **`lab1.sb2`** file that you previously downloaded to your desktop. **Once you see the submission confirmation page** showing that the CS 105 server has your submission saved, you can close your browser and you are finished!

Your TA will randomly choose one of your group's submissions to grade and everyone will receive that grade. ***You should double check that everyone has submitted their lab to the CS 105 server!***