Week 02, Lab 02	Weight: 1%	Due: End of your stream's week 2 lab session (via sync)

Pre-lab Preparation:

- **Week 1 Lectures, Week 1 Lab**
- **Lecture: Wk02 Day 004, 005 Algorithms, Debugging**

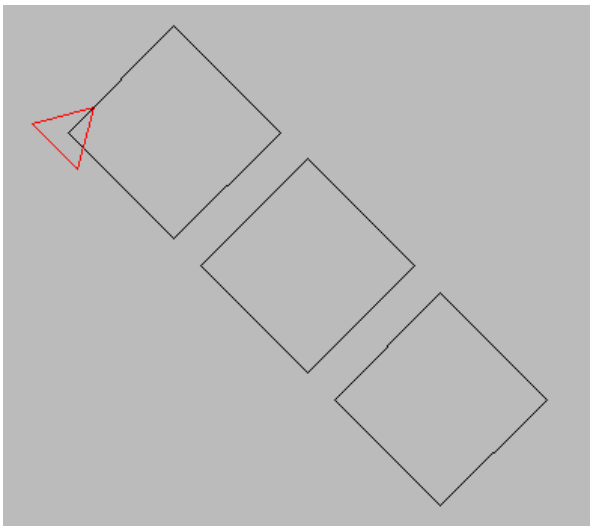
Lab Activities:

Exercise 1: Runtime Error Debugging

Remember to **sync** to obtain the lab starting code.

Navigate to the directory: **`~/p1.2015s1/lab02/working_copy/ex01/`**

Using **`make`**, compile **`lab02ex01.c`** and run the resulting program. Note the program's output. This program should draw the following, *but it does not*:



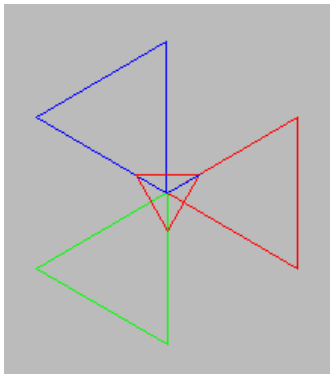
Open the file **`lab02ex01.c`** in the editor, review the code that is present. Make changes to the code to fix the runtime bug which causes the program to not execute correctly. Recompile the source code and ensure your program achieves the desired output, as above.

Also the function comments have only been written for the **`main`** and **`move_to_next`** functions. Write function comments for each of the functions in the **`lab02ex01.c`** source file. Finally accurately complete the file comment at the top of the source file with your details.

Exercise 2: Runtime Error Debugging

Navigate to the directory: **`~/p1.2015s1/lab02/working_copy/ex02/`**

Using **`make`**, compile **`lab02ex02.c`** and run the resulting program. Note the program's output. This program should draw three triangles, one red, one green and one blue, but it does not.



Modify the source code such that the program draws the three triangles, such that one is red, one is green and one is blue. Compile and run your program to ensure you get the desired result.

Add debug **`print_stub`** function calls to the source code such that the program's debug panel displays the text: **`"Drawing Red Triangle"`** immediately before the turtle draws the red triangle. Do the same for the green and blue triangles.

Remember to call the **`turn_on_debug_output`** function to activate the debug panel at an appropriate time.

Write appropriate function comments for each of the functions in the **`lab02ex02.c`** source file. Finally accurately complete the file comment at the top of the source file with your details.

Exercise 3: Stick Figure Person Flowcharts

Log into the Lab PC. Launch the diagramming program Visio. Using Visio create a set of flowcharts which document the process of drawing a stick figure person.

There must be at least two complete flowcharts used to document the algorithm of drawing the stick figure person. A main algorithm, and at least one other function, however you are free to create as many flowcharts as required to clearly document your drawing process.

Once complete, have a TA review your completed exercises 1, 2 and 3 for this lab session. See the end of this document for the review questions.

Exercise 4: Syntax Error Debugging

Navigate to the directory: **`~/p1.2015s1/lab02/working_copy/ex04/`**

Using **`make`**, compile **`lab02ex04.c`**. Note the program's source code does not compile. There are syntax errors which stop the program from compiling.

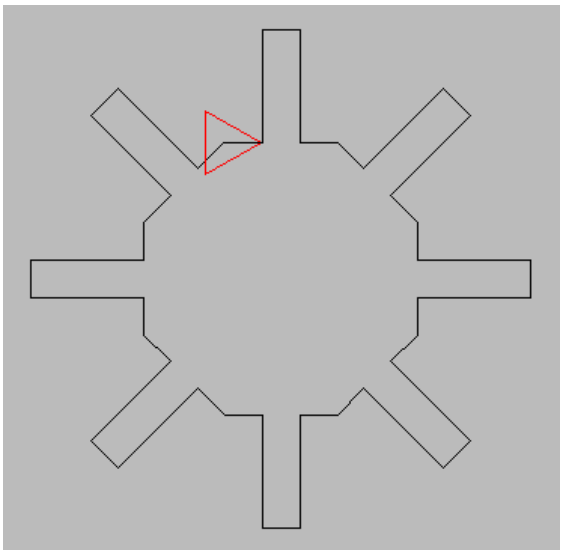
Open the file **`lab02ex04.c`** in the editor, review the code that is present. Make changes to the code to fix the syntax error which causes the program to not compile. Re-run **`make`** to see if you have corrected the syntax errors. If the program successfully compiles, run it. Note the output, does the program now draw three hexagons as required, one blue, one yellow and one magenta?

Write appropriate function comments for each of the functions in the **`lab02ex04.c`** source file. Finally accurately complete the file comment at the top of the source file with your details.

Exercise 5: Using debug print_stub

Navigate to the directory: **`~/p1.2015s1/lab02/working_copy/ex05/`**

Using **`make`**, compile **`lab02ex05.c`** and run the resulting program. Note the program's output. The program should draw the following pattern:



Open the file **`lab02ex05.c`** in the editor, review the code that is present. Add debug **`print_stub`** function calls to each function, use short, high-level English descriptions of what each function does. These **`print_stub`** should help a programmer diagnose which part of the algorithm is currently running, and which hence function is about to be executed, or has finished executing.

Call the **`turn_on_debug_output`** function to activate the debug panel at an appropriate time.

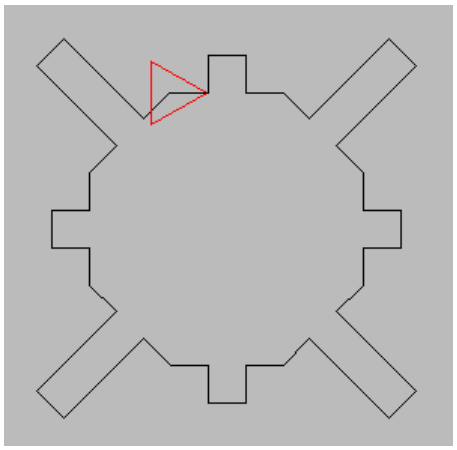
Write appropriate function comments for each of the functions in the **lab02ex04.c** source file. Finally accurately complete the file comment at the top of the source file with your details.

Exercise 6: New algorithm requirements

Copy the completed source code from the previous exercise (**lab02ex05.c**) into the **ex06** directory. Name the newly copied file: **lab02ex06.c** Navigate to the directory: **~/p1.2015s1/lab02/working_copy/ex06/**

Using **make**, compile **lab02ex06.c** and run the resulting program. Ensure the program still runs and draws the pattern as in exercise 4.

Open the file **lab02ex06.c** in the editor, and modify the program such that it draws the following pattern:



You may need to add new functions. Write prototypes for all functions declared in the **lab02ex06.c** source file.

Exercise 7: Function Prototypes

Navigate to the directory: **~/p1.2015s1/lab02/working_copy/ex07/**

Using **make**, try to compile **lab02ex07.c**. The program will not compile, as the main function calls three functions that are not declared before they are used by the **main** function. Find these three functions, and write appropriate function prototypes at the top of the source file for them. After adding the prototypes, ensure the program compiles successfully. Run the program, and see what it outputs. The output should resemble a city skyline.

Next, find the function **draw_sky_tower**, it currently has only one statement inside, a **print_stub** function call. Add to this function turtle commands to draw a Sky Tower-like building. Add any improves the city skyline drawing to enhance the overall program.

Week 02, Lab 02 Submission:

Run the **sync** command to submit your completed lab work.

Shutdown your Raspberry PI by pressing **ALT-CTRL-DEL**. Power-down and pack up your Raspberry Pi kit.

Marking Criteria:

Have you completed each of the following? Have you submitted your code from lab?

Marking Criteria:	Week 02 Lab 02 Weight 1%	Yes	No
Ex 1:	draw_square function corrected?		
	Function comments completed?		
	File comment completed?		
Ex 2:	Three correctly coloured triangles drawn?		
	Required print_stub calls added?		
Ex 3:	Stick figure flowcharts designed and detailed?		
Ex 4:	Code compiles successfully?		
	Three correctly coloured hexagons drawn?		
Ex 5:	Debug print_stub calls added?		
	Function comments added?		
Ex 6:	Correct pattern drawn?		
	Additional useful functions created, along with function prototypes?		
Ex 7:	Function prototypes added?		
	draw_sky_tower function logic defined such that a Sky Tower-like building is drawn?		
	Total:		

Next activity: Final Week 2 Lecture and Week 2 Homework.