Introduction to the C Compiler

Lab 2

Section B

Mitchell Wadle

Date: 09/08/17

Problem:

In this lab we were asked to create simple programs that performed a task and compile them to get used to the complier and compiling our code. The first code what just a shell of the other codes. The next code simply printed a string text on the screen. The next code we modified the previous string to read our name, class and date. The next code we made was a simple program that asked for a width and a height of a rectangle and returned what the area of that rectangle would be. The last code was we had to modify the code to take in width, height, and length of a rectangular prism and return the volume of the prism. The overall objective of the lab was to introduce us to compiling code and creating simple performing codes.

Analysis:

For the first few codes we only had to copy what the lab instructions said. It wasn't until the last problem where we were asked to add to the code for it to perform a different task. They wanted another variable, z, and the formula for the area needed to be changed to find the volume of a rectangular prism.

Design:

I changed the code in the following order:

1. Add z to int x, y
2. Add another printf that reads ("enter a length: ")
3. Add scanf after to read a value for z
4. Modify the final statement to read ("A prism that is %d by %d by %d has a volume of %d")
5. Finally modify the end so that z is entered to the third %d and the formula should be changed from x*y to x*y*z.

Extra comment lines were also added and the code was separated out of one could better read and understand the code and what it does.

Testing:

After creating each code, the code was compiled and run in Cygwin to verify that the code does indeed work and has no errors. Every test I ran in Cygwin printed/perform what was needed.

Comments:

This lab did a good job of showing us the compiler and how it prints/performs the code in Cygwin. The code we were asked to create was rather simple and easy to understand and modify. The problem was resolved and this was the first lab to ask us to alter the code to perform a different task without telling us exactly how.

Source Code:

Lab2-1.c

```
//Mitchell Wadle, Section B, 09/08/17
//LAB2-0.c : Defines the entry point for the console application.

#include <stdio.h>

int main(int argc, char* argv[]){
	printf("Mitchell Wadle CprE 185 09/08/17\n");
	return 0;
}
```

Lab2-2.c

```
//Mitchell Wadle, Section B, 09/08/17
//LAB2-0.c : Defines the entry point for the console application.

#include <stdio.h>

int main(int argc, char* argv[]){
	int x, y;

	printf("Enter a width:");
	scanf("%d", &x);
	printf("Enter a height:");
	scanf("%d", &y);
	printf("A %d by %d recangle's area is %d\n", x,y,x*y);
	return 0;
}
```

Lab2-3.c

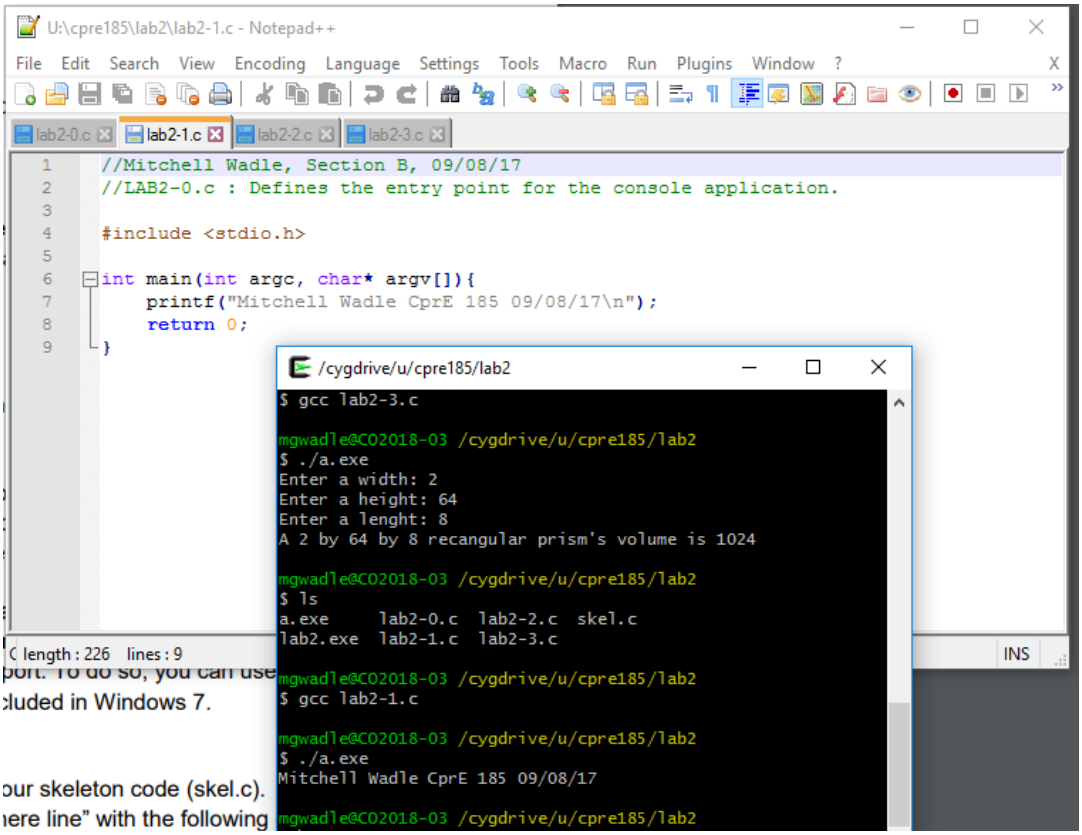
```
//Mitchell Wadle, Section B, 09/08/17
//LAB2-0.c : Defines the entry point for the console application.

#include <stdio.h>

int main(int argc, char* argv[]){
	int x, y, z;

	printf("Enter a width: ");
	scanf("%d", &x);
	printf("Enter a height: ");
	scanf("%d", &y);
	printf("Enter a length: ");
	scanf("%d", &z);
	printf("A %d by %d by %d recangular prism's volume is %d\n", x,y,z,x*y*z);
	return 0;
}
```

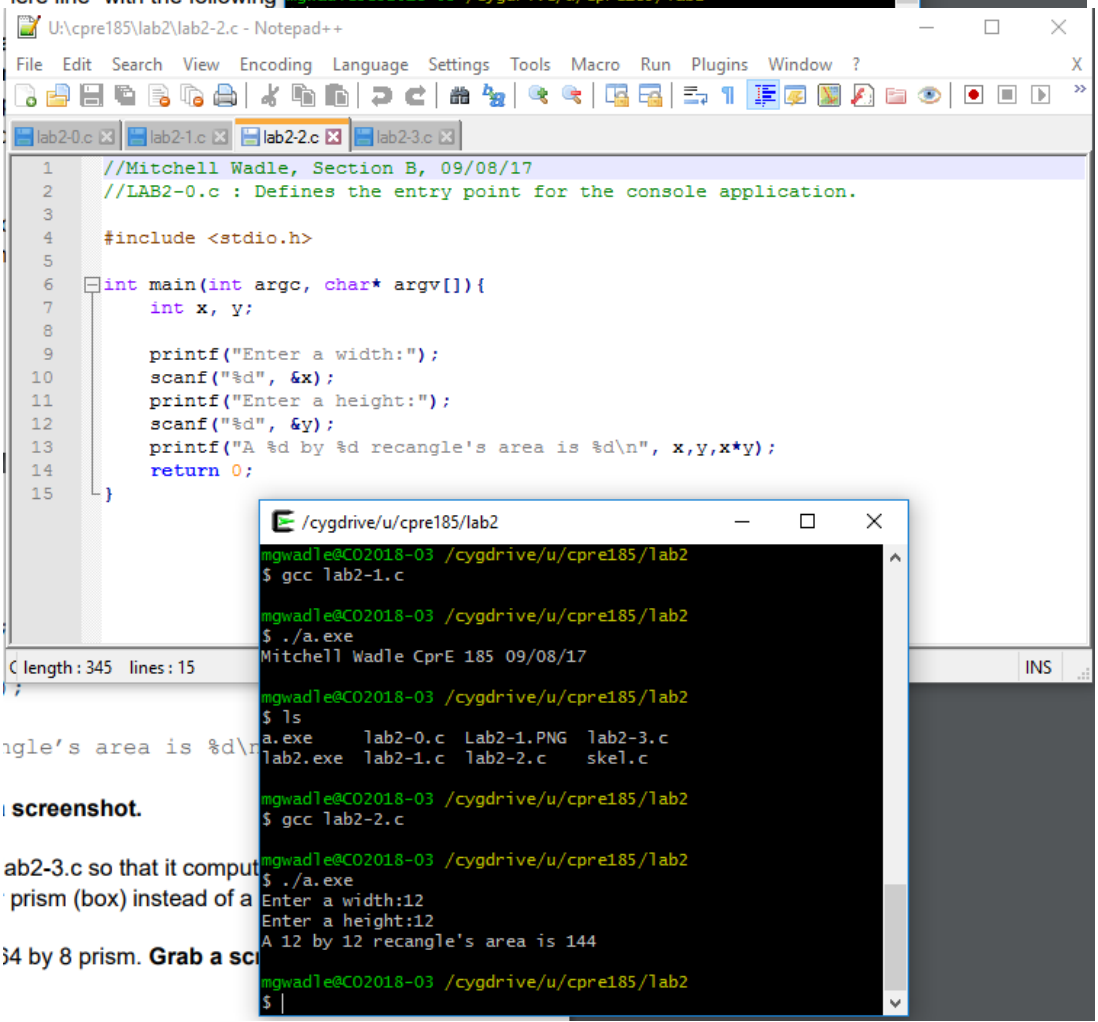
Screenshots:



cluded in Windows 7.

our skeleton code (skel.c).

here line" with the following



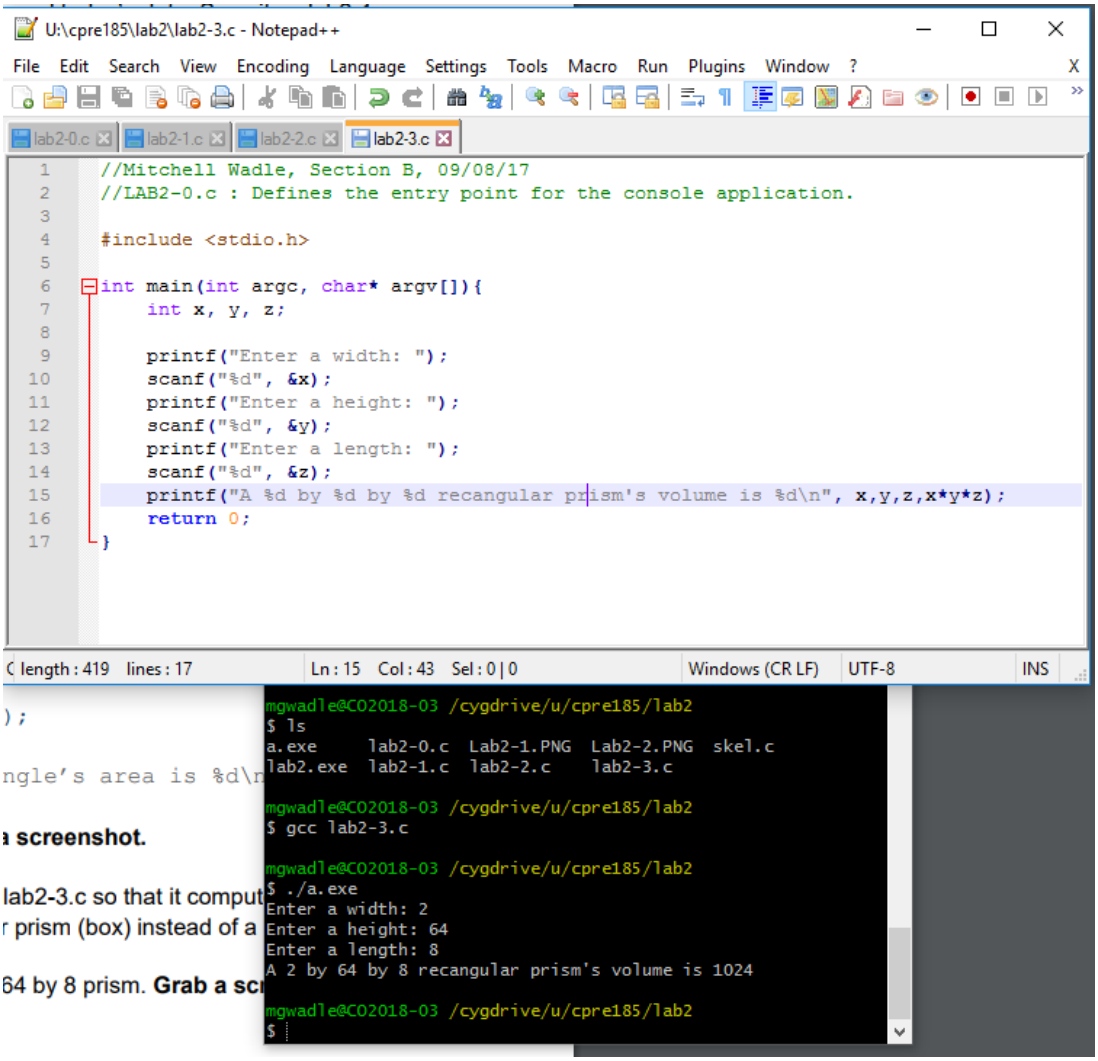
ngle's area is %d\n

screenshot.

ab2-3.c so that it comput

prism (box) instead of a

54 by 8 prism. **Grab a sc**



) ;

ngle's area is %d\n

a screenshot.

lab2-3.c so that it comput
r prism (box) instead of a

64 by 8 prism. **Grab a sc**