CodeBlocks is already installed in the lab. The instructor will explain how to set up a project in, how to write code, compile, build, execute, etc.

- If you are doing this by yourself, make sure you follow the directions in the file *__notes_installing_CodeBlocks*, available in Canvas.

The instructor will also explain what and how to submit the lab report. All submissions are electronic, in Canvas!

The purpose of this lab is just to get used to typing C code exactly – do not worry if you do not understand all the parts of each program!

[0] Warmup: What is the error in this code? Which line?

```
1: #include <stdio.h>
2:
3: int main(void)
4: {
5:    printf("Hello, World!")
6:    return 0;
7: }
```

Answer: *There is no semicolon on line five after the print statement*

[1] Enter the following program, compile it, and execute it:

```
#include <stdio.h>
int main() {
    printf( "Hello, world!" );
    printf( "C programming\n" );
    printf( "at Tarleton\n" );
    return 0;
}
```

Explain in your own words what the program does (did you notice the newline characters \n?):

The first two printf statements blend together on the same line because there is no \n on the first printf statement.

[2] Solve Exercise 2 at the end of Lesson 1:

Enter the following program, compile it, and execute it:

```
#include <stdio.h>
int radius, area;

int main( void )
{
   printf( "Enter radius (i.e. 10): " );
   scanf( "%d", &radius );
   area = (int) (3.14159 * radius * radius);
    printf( "\n\nArea = %d\n", area );
    return 0;
 }
```

Explain in your own words what the program does:

The program asks for a number from the user and the program then calculates the area of a circle but only returns the whole number (or only the integer of the number) and returns said integer

Answer the questions:

- What lines contain statements? *6, and 9*
- What lines contain variable definitions? *2, and 7*
- What lines contain comments? *None*
- On what lines are functions called? *None*

[3] Solve Exercise 3 at the end of Lesson 1:

Enter the following program, compile it, and execute it:

```
#include <stdio.h>

int x, y;

int main( void )
{
   for ( x = 0; x < 10; x++, printf( "\n" ) )
       for ( y = 0; y < 10; y++ )
           printf( "X" );

   return 0;
}
```

Explain in your own words what the program does:

The program makes a grid of a 10x10 filled with "X"

[4] Count in decimal, binary and hex from 0 to 33 (side-by-side):

Dec	0	1	2	3	4	5	6	7	8
Bin	0000	0001	0010	0011	0100	0101	0110	0111	1000
Hex	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8

9	10	11	12	13	14	15	16	17	18
1001	1010	1011	1100	1101	1110	1111	0001 0000	0001 0001	0001 0010
0x9	0xA	0xB	0xC	0xD	0xE	0xF	0x10	0x11	0x12

19	20	21	22	23	24	25	26	27	28
0001 0011	0001 0100	0001 0101	0001 0110	0001 0111	0001 1000	0001 1001	0001 1010	0001 1011	0001 1100
0x13	0x14	0x15	0x16	0x17	0x18	0x19	0x1A	0x1B	0x1C

29	30	31	32	33	34	35	36	37	38
0001 1101	0001 1110	0001 1111	0010 0000	0010 0001	0010 0010	0010 0011	0010 0100	0010 0101	0010 0110
0x1D	0x1E	0x1F	0x20	0x21	0x22	0x23	0x24	0x25	0x26

To submit for lab report

[1] Solve Exercise 2 at the end of Lesson 2:

Enter the following program, compile it, and execute it

```
1:  /* ex02-02.c */
2:  #include <stdio.h>
3:
4:  void display_line(void);
5:
6:  int main(void)
7:  {
8:      display_line();
9:      printf("\n Teach Yourself C In One Hour a Day!\n");
10:     display_line();
11:
12:      return 0;
13: }
14:
15: /* print asterisk line */
16: void display_line(void)
17: {
18:     int counter;
19:
20:     for( counter = 0; counter < 30; counter++ )
21:         printf("*" );
22: }
23: /* end of program */
```

- Attach screenshots of source code and output

```
[Running] cd "c:\Users\austi\OneDrive\School\Electrical\Procedural
Programming\Lab\Lab 1\" && gcc Lab_1.c -o Lab_1 &&
"c:\Users\austi\OneDrive\School\Electrical\Procedural Programming\Lab\Lab 1\"Lab_1
******************************
 Teach Yourself C In One Hour a Day!
******************************
[Done] exited with code=0 in 0.424 seconds
```

Explain in your own words what the program does:

Puts stars on the lines before and after the print statement "Teach Yourself C In One Hour a Day!"

Answer the questions:

- What lines contain statements? *9, 21*
- What lines contain variable definitions? *4, 6, 18*
- What lines contain function prototypes? *4*
- What lines contain function definitions? *6, 16*

- What lines contain comments? *1, 15, 23*
- On what lines are functions called? *8, 10*