# Exercise - Digital Input

Read through and try to understand the following two tutorials in preparation for the programs that you will need to write later on in this lab.

Tutorial 1: Digital Read Serial
Tutorial 2:Button

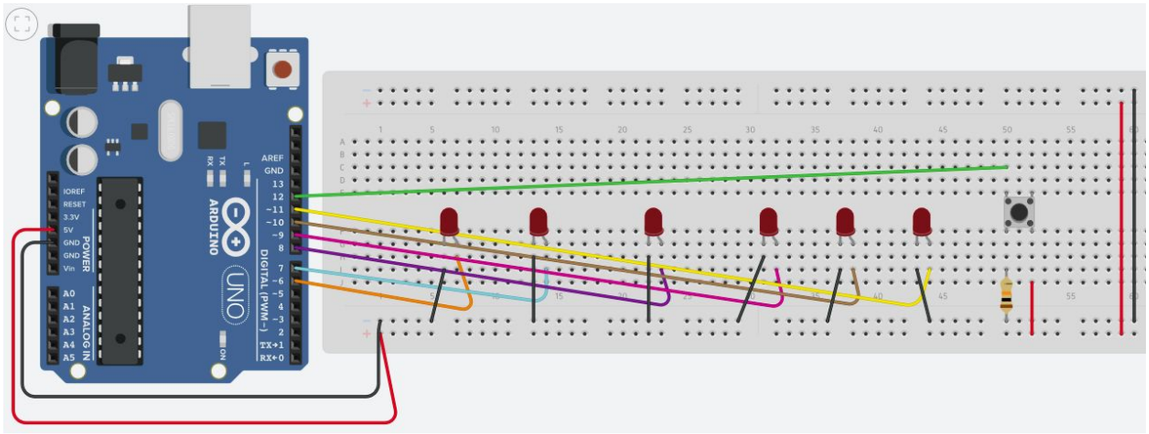
For the two programs you need to write below, you need to have a basic knowledge of how if-else if- else statements work. Browse through these links in order:

- ➢ Basic if statement note
- ➢ if-else if-else Note
- ➢ AND operator
- ➢ OR operator
- ➢ NOT operator

You also need to understand how to write your own Arduino functions. You are not restricted to only using void loop() { } and void setup() { }. Read through the reference document here: https://www.arduino.cc/en/Reference/FunctionDeclaration (We have covered this concept in lab #1)

You will also need to read through the two posted PowerPoints on Selection Structures.

**CIRCUIT SETUP:** You will wire six LED's in total ranging from digital I/O pins 6 to 11. Remember that each LED needs a resistor to protect it. Wire your push button (see the picture below) and connect it to pin #12. You will need to ground the push button pin using a 10k Ohm resistor.



**SCENARIO:**
When the push button is pressed, randomize a number from 1 to 6 from inside your main loop() function. If the number 2 is chosen then the first two LED's from left to right should turn on for exactly five seconds. After the time is up, turn off all LED's. If the number 5 is chosen then all five LEDs from left to right should turn on for exactly five seconds. After the time is up, turn off all LED's. Make your own function to turn off all LED's.

Create your own function called: ***void turnOnLEDs(int number) { }*** where the input parameter ***number*** is the randomly generated number you get. You pass the number generated as a parameter into the new function you have just created. Inside the function, you will have If statements so that the correct number of LED's turn on.

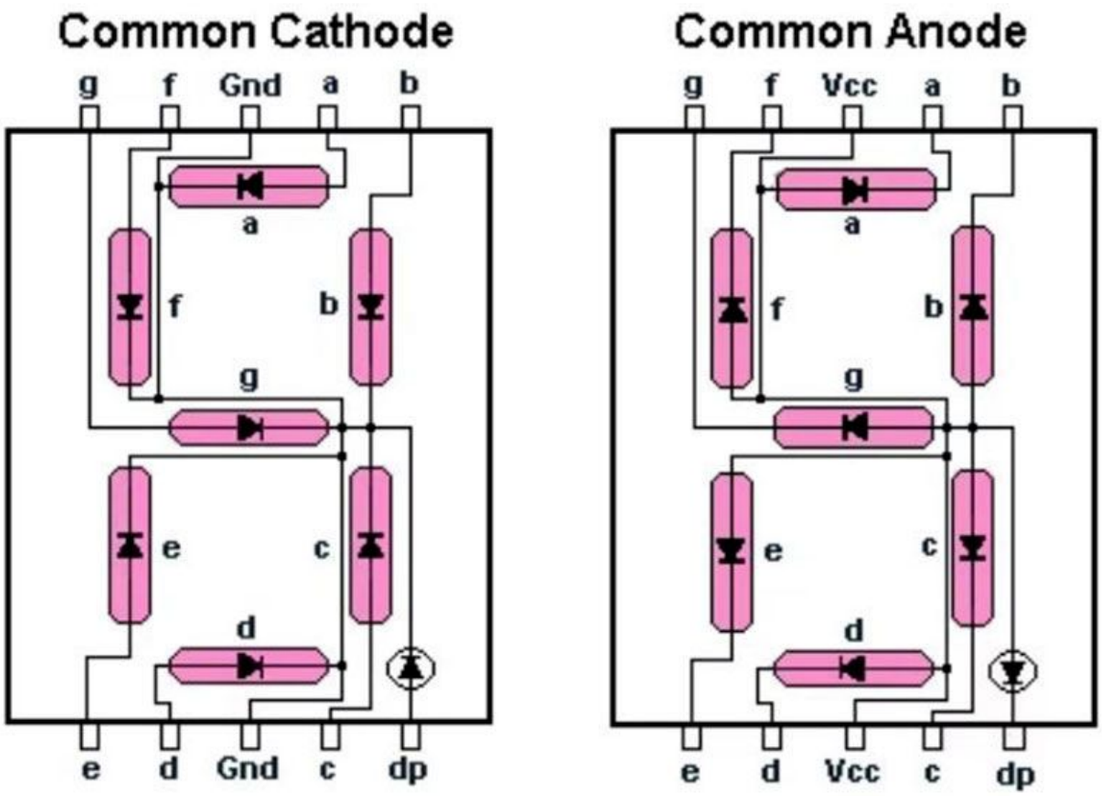
```
void turnOnLEDs(int number)
{
}
```

Use Serial.println() statement inside the function to print out the number generated to the Serial Monitor. Remember to comment your code fully before submitting your file.
**>> Call your program *YourFirstName_Input.ino***

# Exercise - Digital Input

**Extension:**

**Learn how a seven segment (common cathode) display works.**
**The idea of the original code works stays the same. You will press the push button and generate a number between 1 and 6. The difference is that you will be showing the actual number on the display. Create a new Tinkercad sketch. You will only need a breadboard, an Arduino, a seven segment display (common cathode), and a push button.**




*SSD Configuration*

Gnd represents GROUND.
Picture taken from:
https://www.digikey.com/en/maker/projects/how-to-interface-a-seven-segment-display-with-an-arduino/9c05f147618c4fe3b8bb79acce5c60e3