



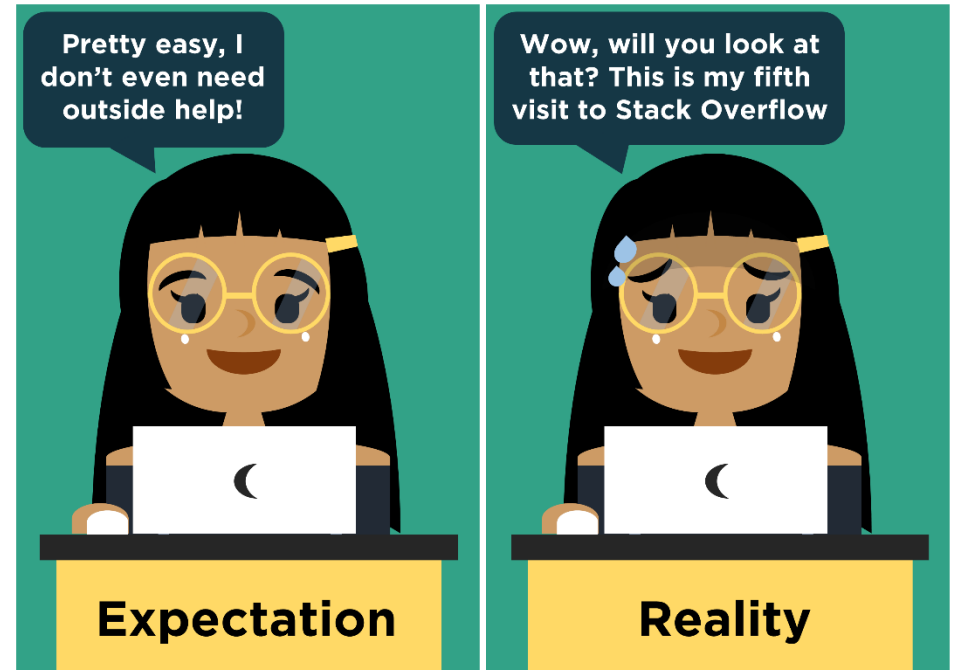
Data Driven Design

Topic 1  
Microcontroller Programming  
Instructor: Jay Dhariwal,  
Asst. Prof.,  
Dept of Design,  
IIT Delhi  
12<sup>th</sup> January 2024

# Programming

- Programming – instruction to perform task
- English or Hindi – grammar, similarly programming languages have syntax
- Activity: how would a calculator add two numbers?

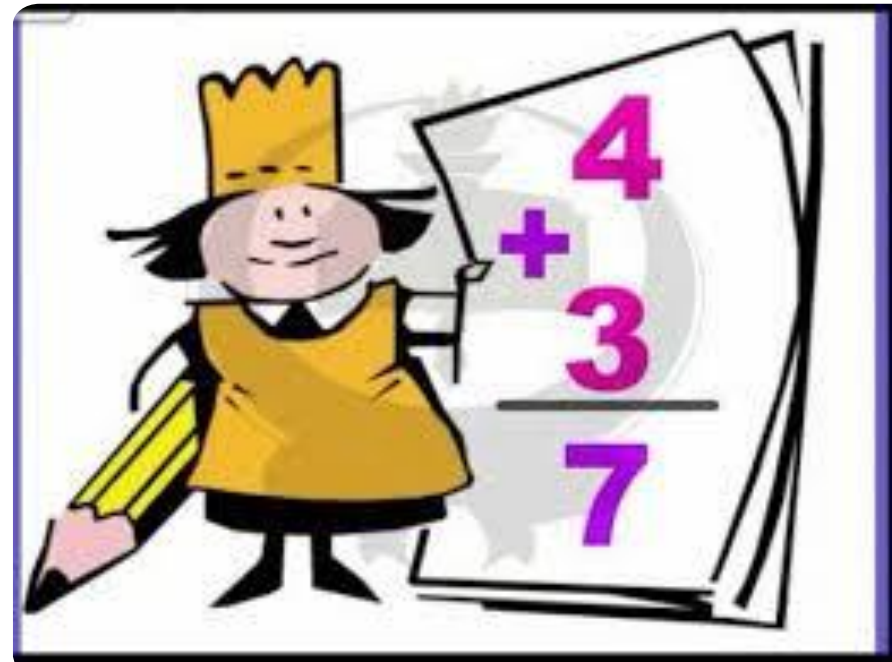
“Programmers: Expectation VS Reality”



fb: Guen's Comics tw/ig: @guenscomics

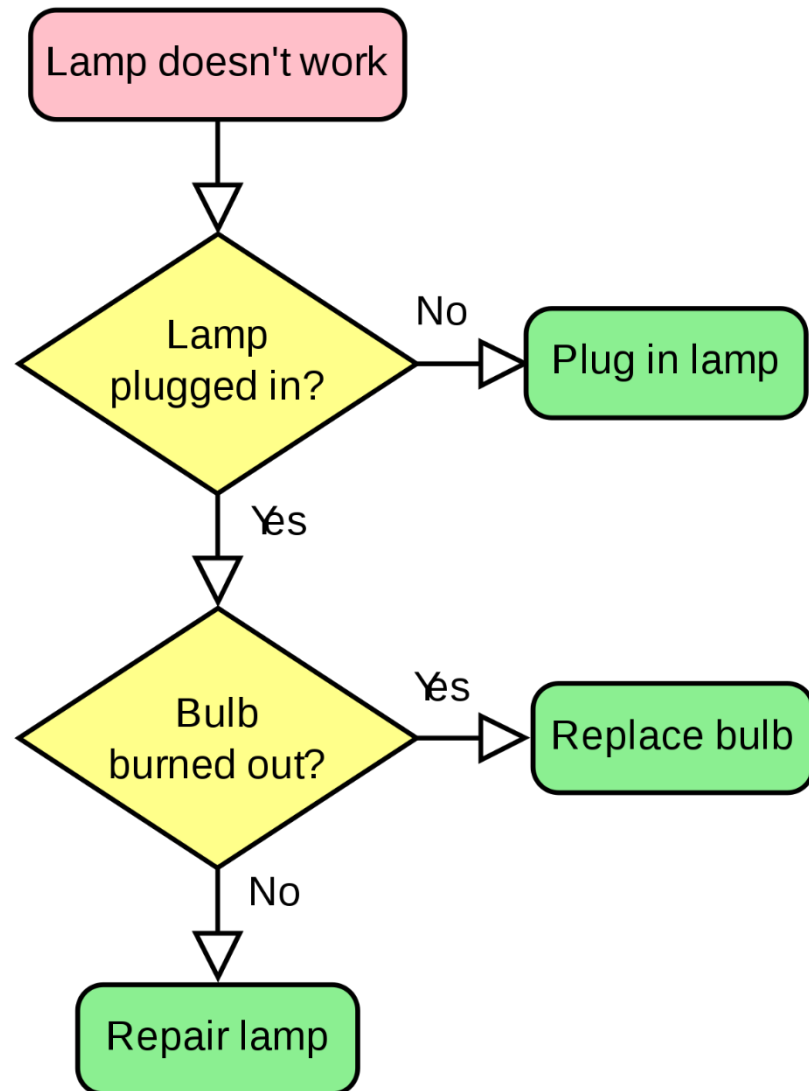
# Algorithm for adding two numbers

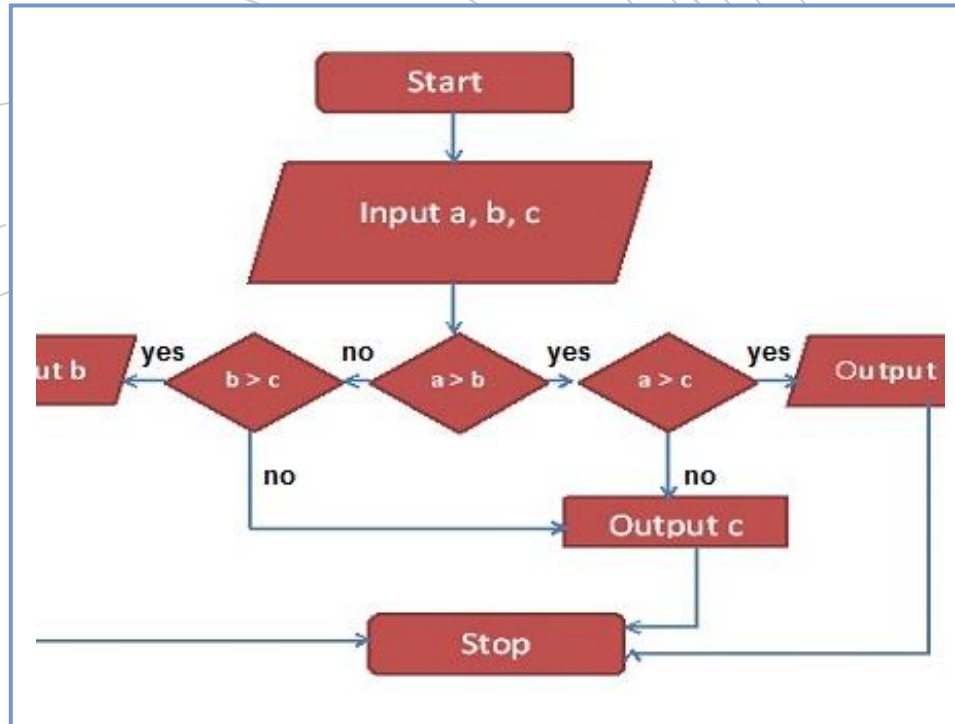
- Input number 1
- Input number 2
- Add number1 and number 2
- Print the result.



# Flowchart

- Languages have different syntax but are largely similar.
- Learn one language and you can learn others quickly.
- Flowchart/Algorithm is the key.
- Activity: Flowchart for repairing a lamp that doesn't work?





Flowchart for largest of three numbers

Programming Language	Application
C	Microcontroller/ Embedded programming, Efficient at runtime. 95% embedded programming in C.
Python	One of the best to teach programming, Scientific computations, Raspberry Pi. Efficient in development times.
Javascript	Creating web pages. Run in browsers. HTML+CSS+Javascript
Scratch	Graphical language, flowchart based for children. MIT App Inventor related to it.
Processing	GUI for Arduino
Visual Basic	Windows based, Event based programming, Easy to build GUI, VBA in Excel.
.NET	Software framework from Microsoft

# C programming

## My first program in C

```
#include <stdio.h>
int main()
{
    printf("Hello World!");
    return 0;
}
```

- [Online compiler](#)
- [C for beginners](#)
- [Examples](#)



Basic  
elements of  
programming  
language

---

Programming Environment

---

Data Types, Variables, Keywords

---

Input and Output Operations

---

Logical and Arithmetical Operators

---

If else conditions, Loops

---

Functions

---

Comments, Indentation, Bottom up debugging



# Python

## Python Advantages & Disadvantages



ADVANTAGES



DISADVANTAGES

- [Python basics](#) from Sanju Ahuja
- [Python interpreter](#)
- [Python examples](#)
- [Fab Academy tutorial on python](#)

# Scratch



- [Scratch animations](#)
- Scratch games
- Arduino with Scratch



**MIT**  
APP INVENTOR

# Processing IDE

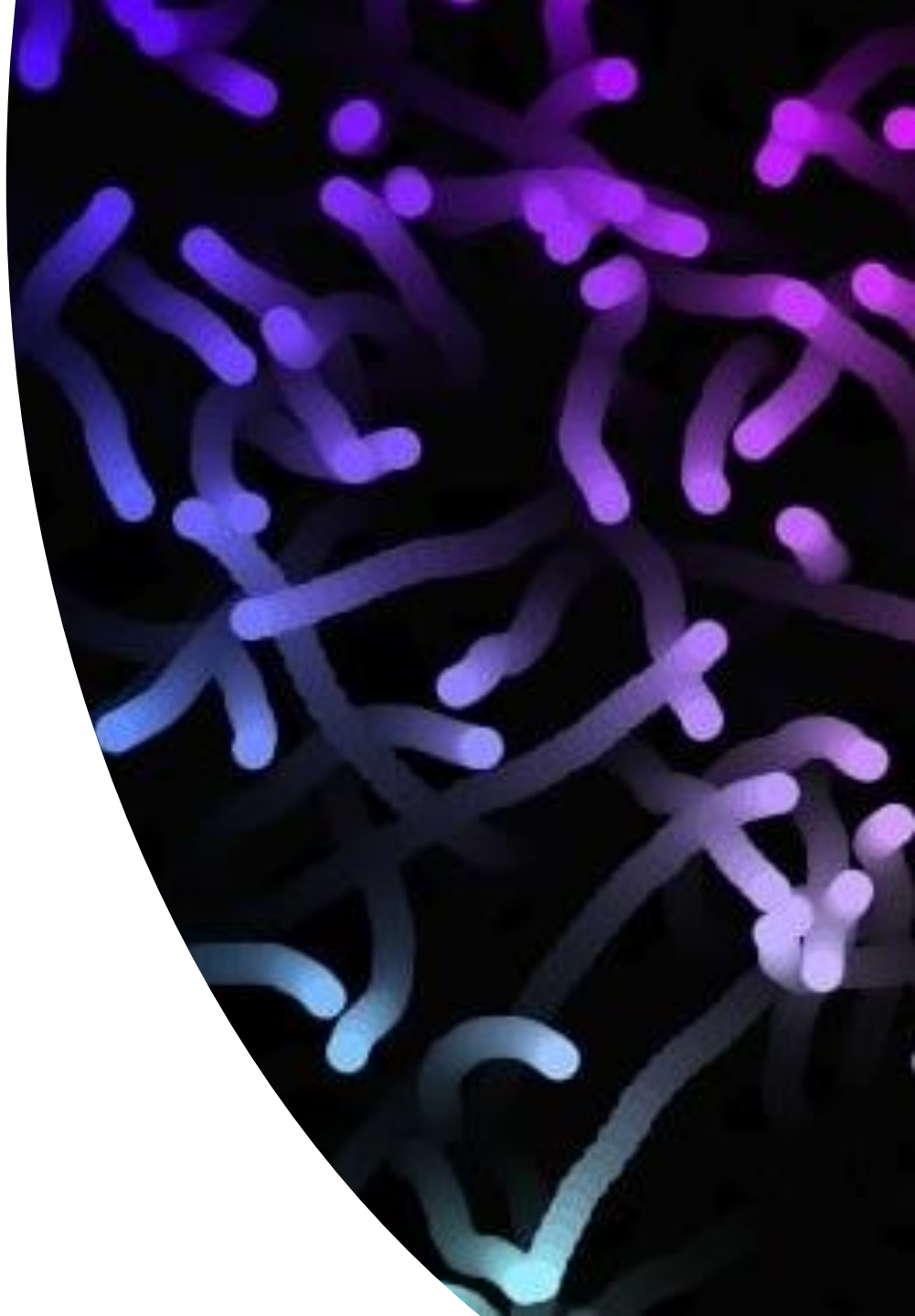
- [Graphical programming language](#)  
–visual design, images, creative applications
- [Youtube tutorials](#)
- Extendable through libraries (written in Java)
- Use for creating GUI for Arduino projects. For visualizing the output from sensors.
- [Other applications](#): Motion graphics, Data visualization, [Music visualization](#)



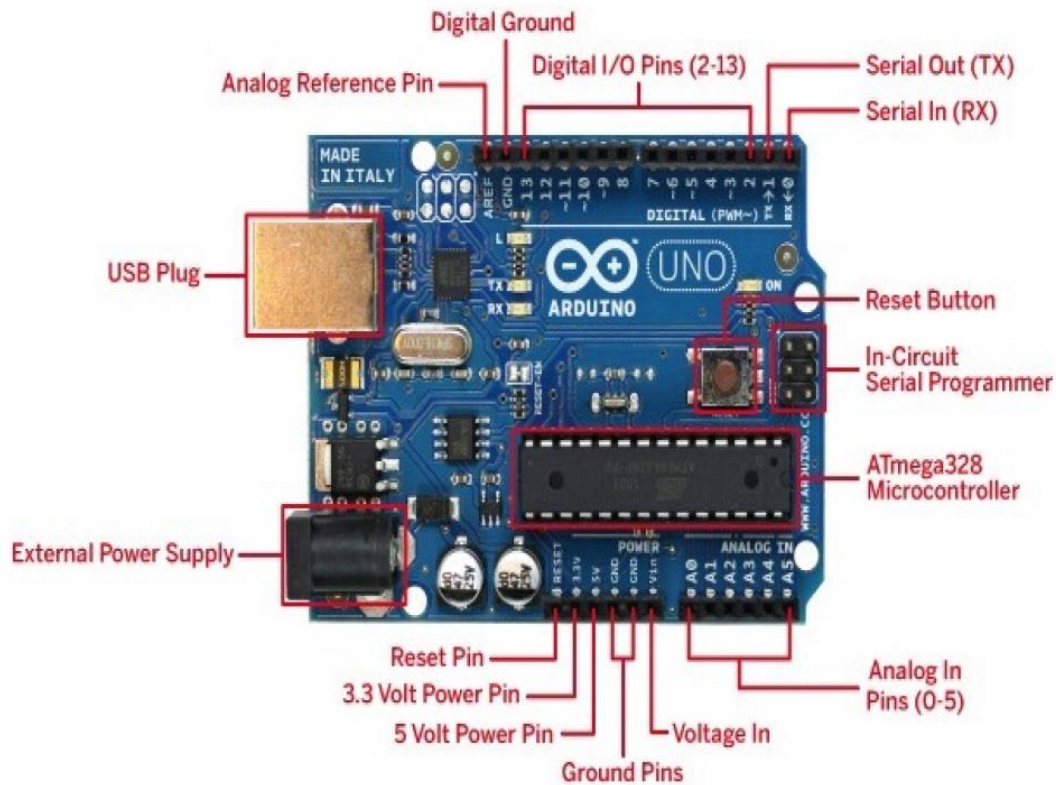
# Processing examples

---

- [Writing simple programs](#)
- [Processing functions reference](#)
- [Examples](#), import libraries
- [Creative, Fun programming](#)
- Setup, draw, events, random, other functions, [rendering a processing sketch](#)
- [Processing for Android](#)



# Microcontroller programming

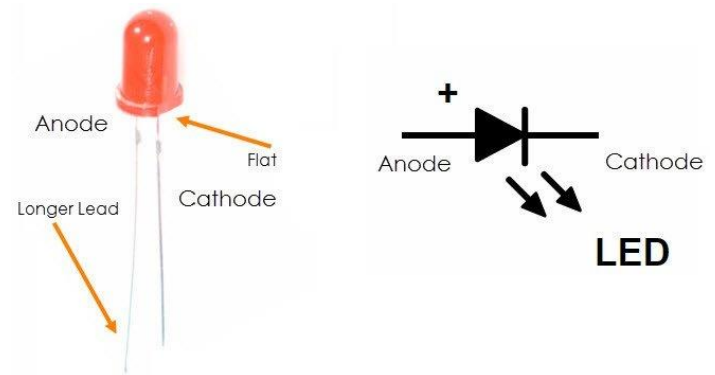
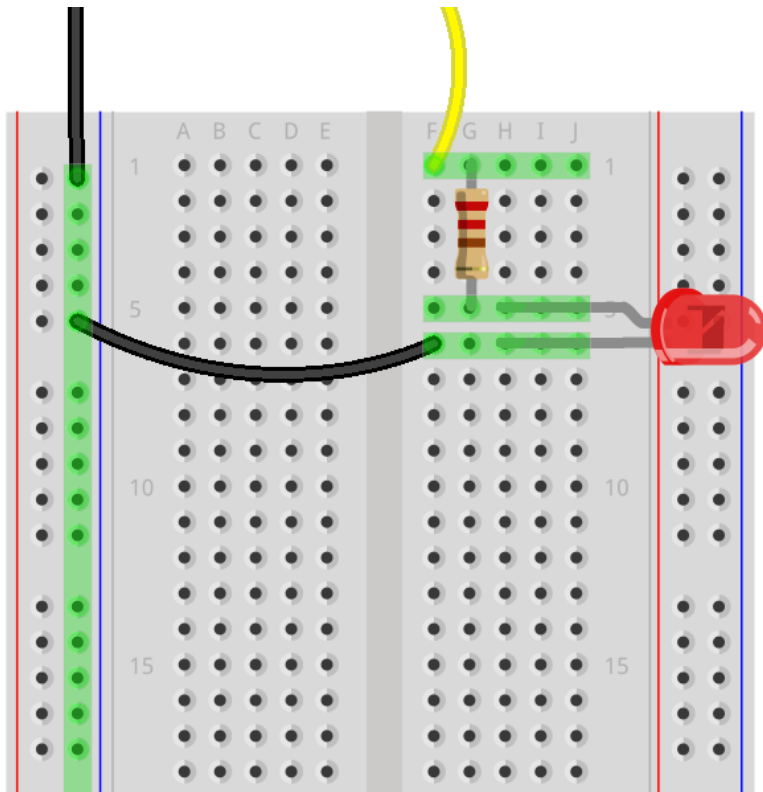


- [DIY devices](#)
- Arduino hardware
- Arduino IDE
- Arduino libraries

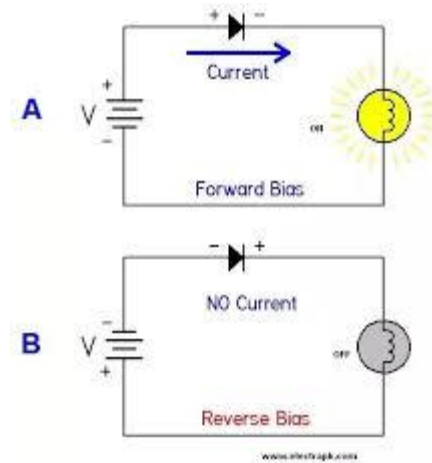
# Getting started with Arduino

- Download [Arduino IDE](#)
- [Arduino kit pdf](#) for instructions
- Part 1 Preparation
- Open Arduino IDE
- Blink sketch with pin 13
- Blink sketch initialize pin 13





Diode is like a one-way valve of electronics

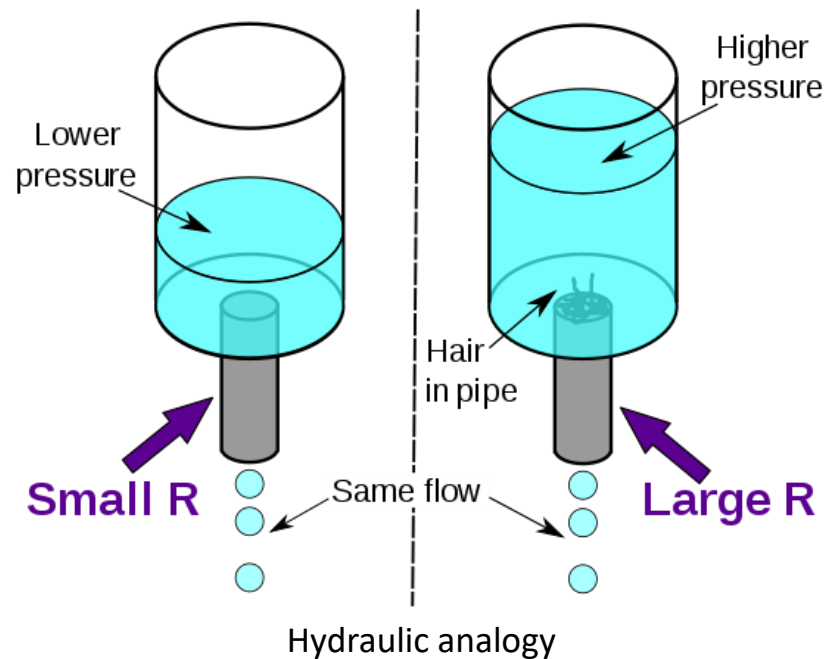


LED is a diode

- LED is a diode (which makes the current flow only in one direction).

# Blinking LEDs

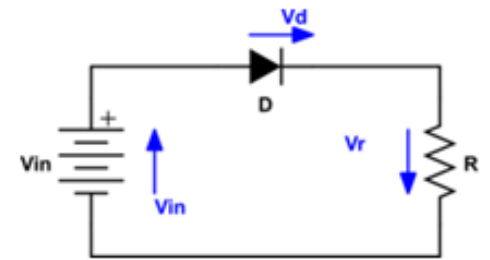
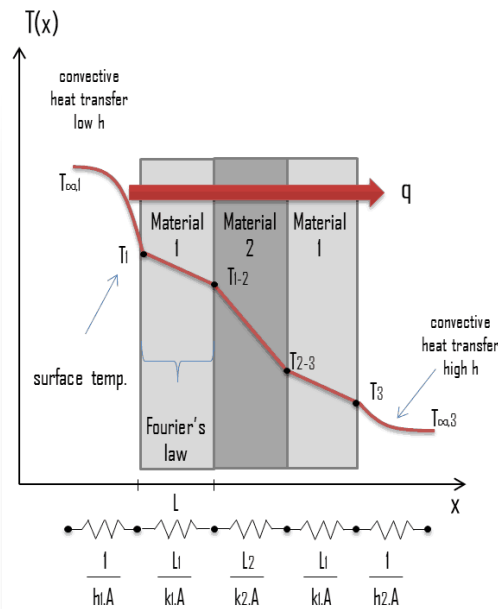
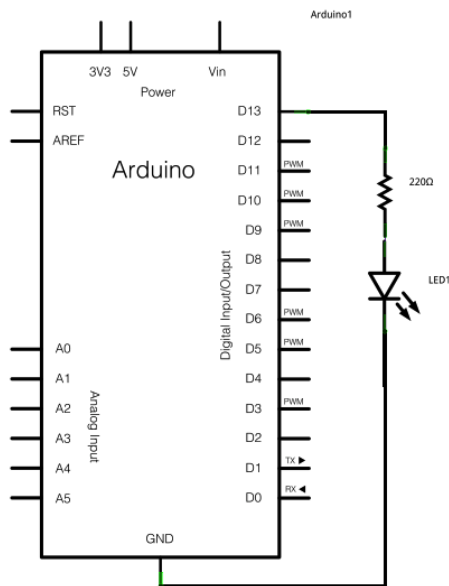
- Part 2 Module Learning
- 2.1 LED
- Blink sketch (pin 12) with own LEDs and [resistors](#)



Ohm's Law,  $V=IR$



# Thermal analogy, $Q = dT/R$ , $I = V/R$

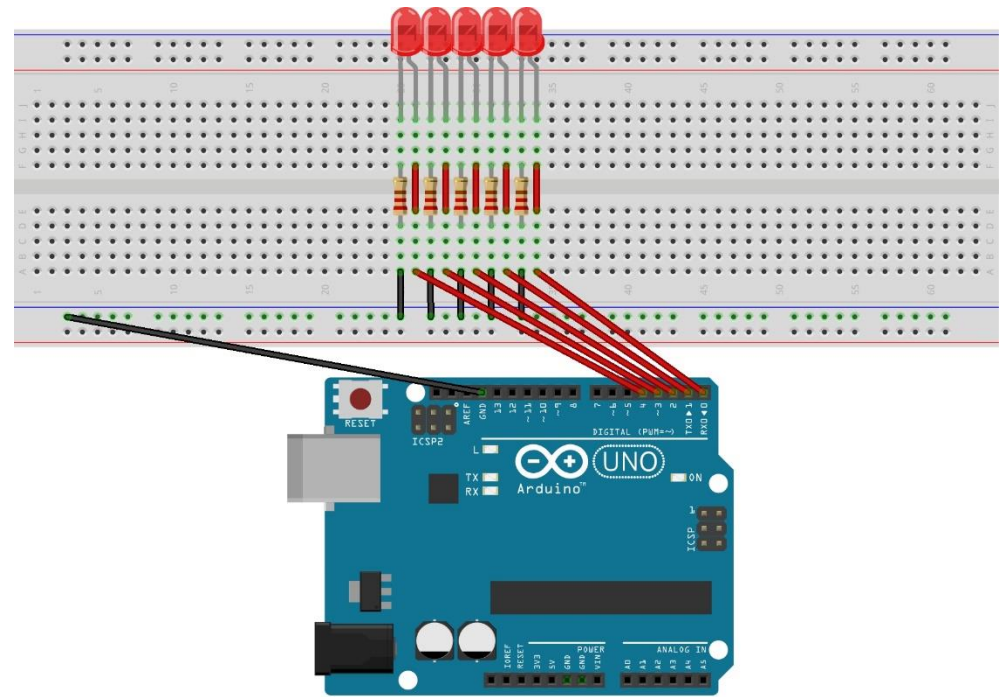


$$\sum_{k=1}^n V_k = 0$$

Kirchoff's Voltage Law

# Blinking LEDs

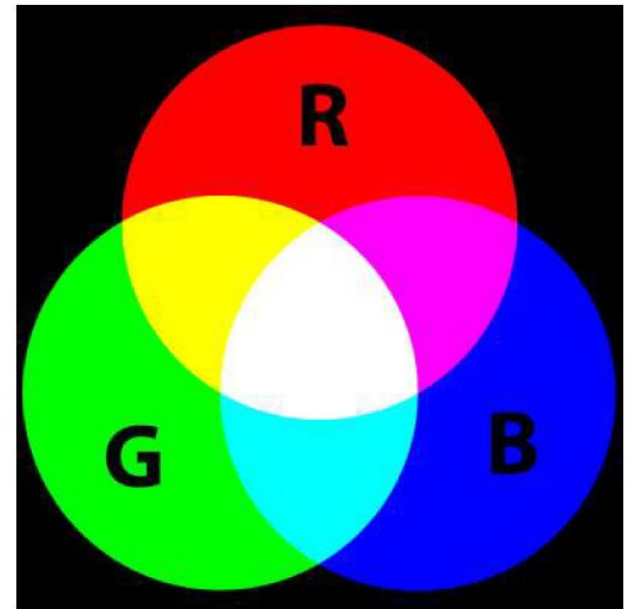
- Blink sketch (pin 12) with own LEDs and resistors
- Blink sketch (pin 12) with own LEDs and resistors and pin 13. (Sequentially on and off vs. Both on and Both off.)

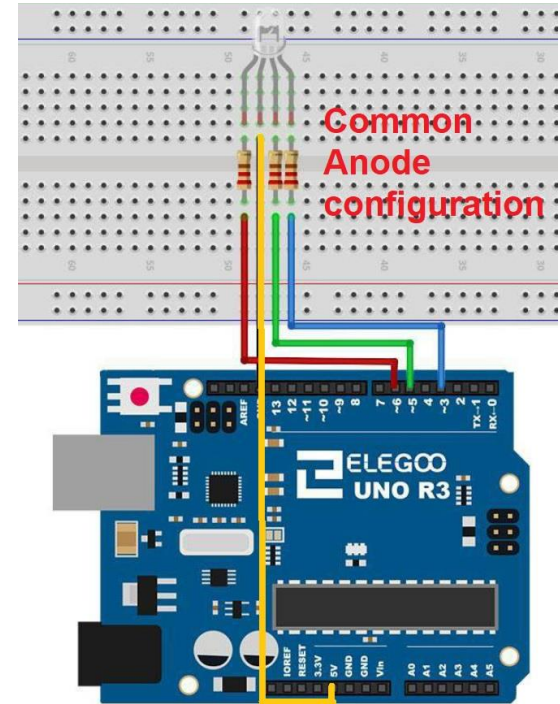
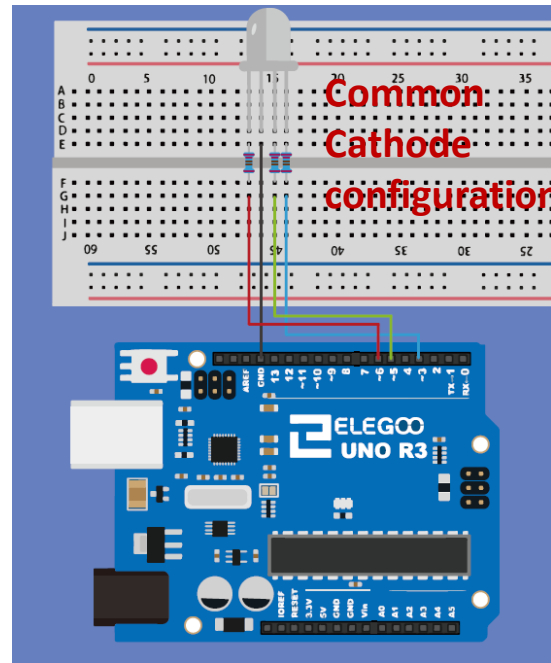
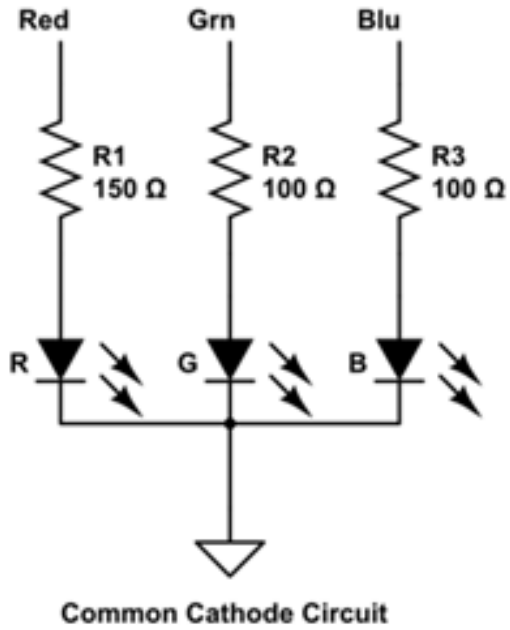


# RGB LED

- PWM with RGB LED with common cathode (Lesson 2.2). NOTE: Longest leg of RGB LED goes to ground pin.
- RGB code

[RGB basic RGB favorite color](#)

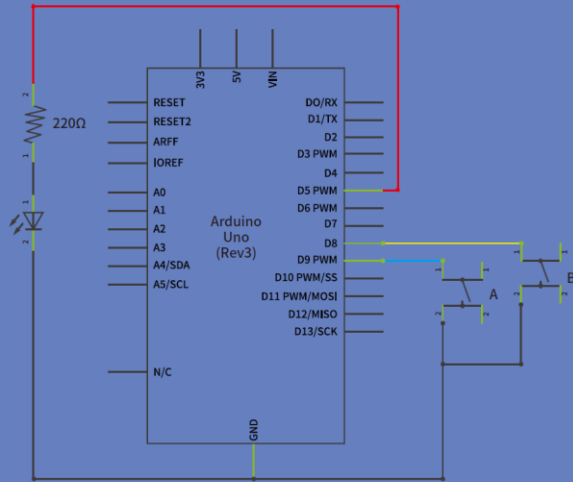




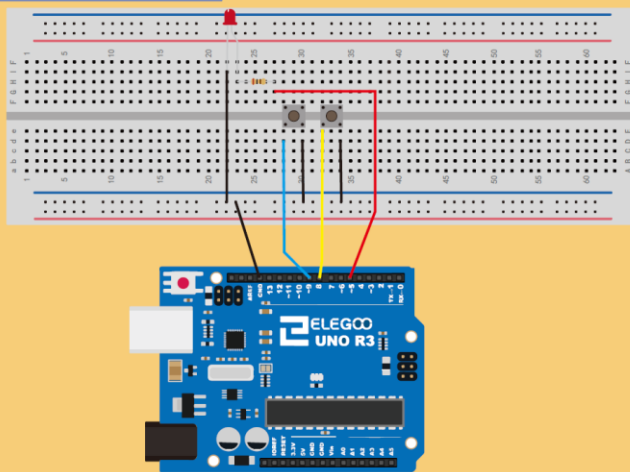
# RGB LED circuit

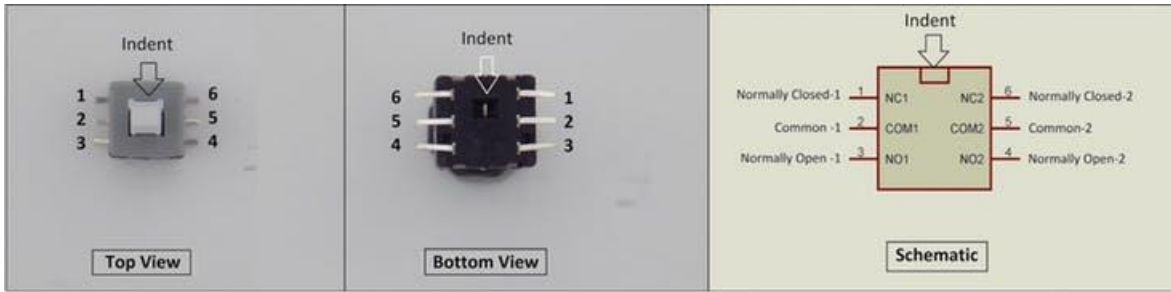
# Push buttons (Input Pullup) – Lesson 2.3

Connection Schematic:

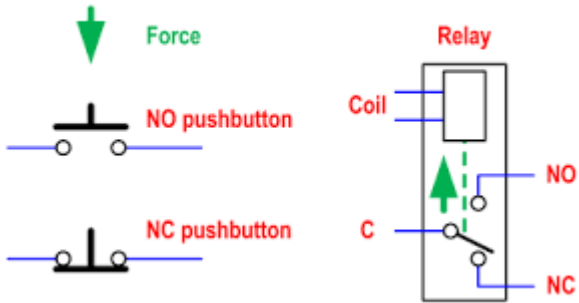
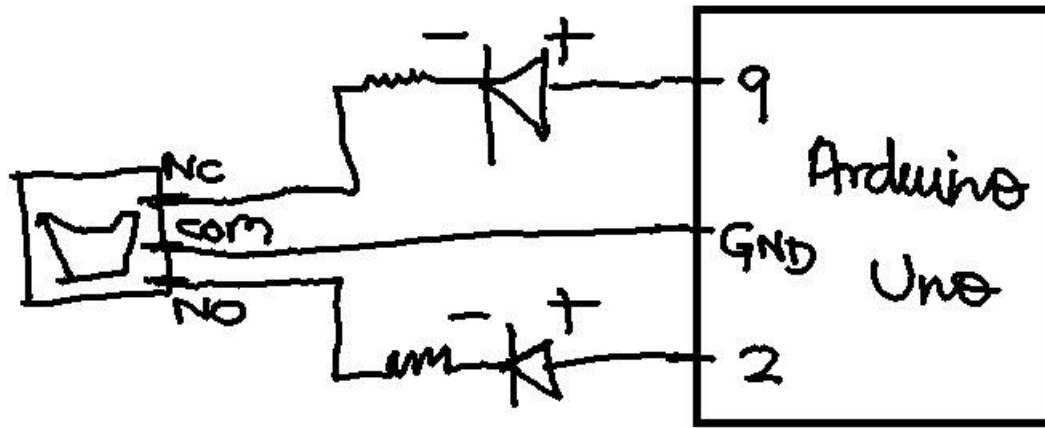


- By default, input pullup pin is High, when push button pressed, it gets to Low. [Video](#)





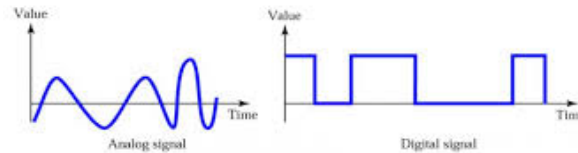
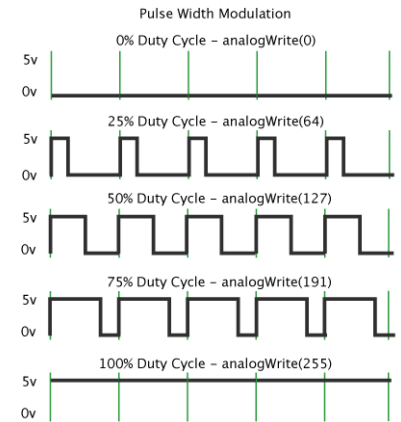
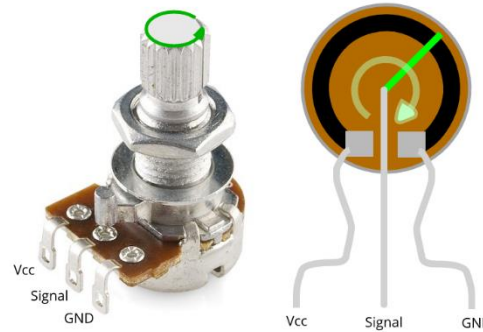
6 Pin Push Switch (Mini DPDT Push Switch) Pinout



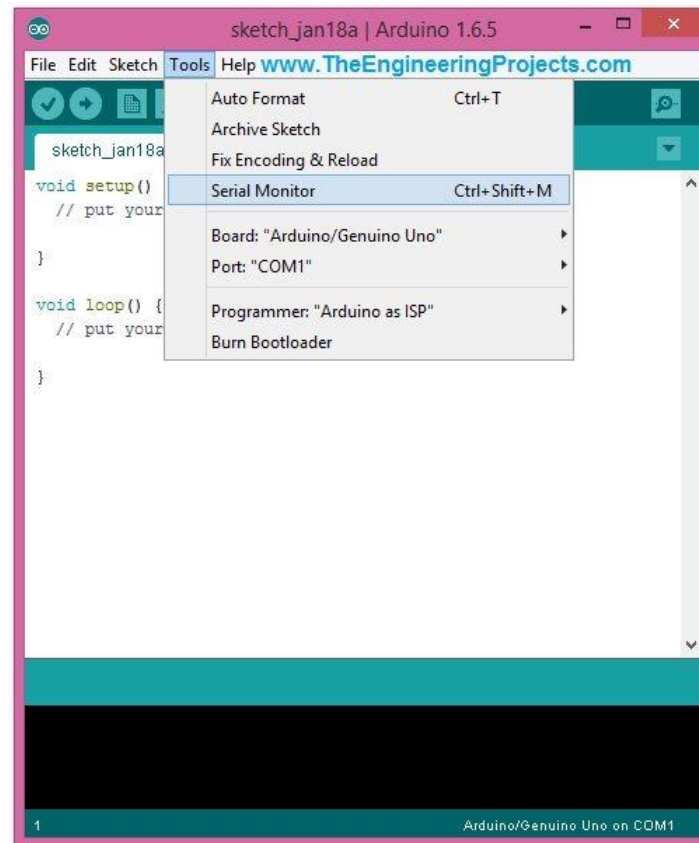
- [DPDT switch video](#)

# Digital vs. Analog signal

- digital output = Blink sketch
- digital input\_pullup = push button example, Lesson 2.3
- analog output = Fade example, PWM pins (~)
- Analog input = AnalogInOutSerial (potentiometer)
- Multimeter = voltages, resistors, continuity.

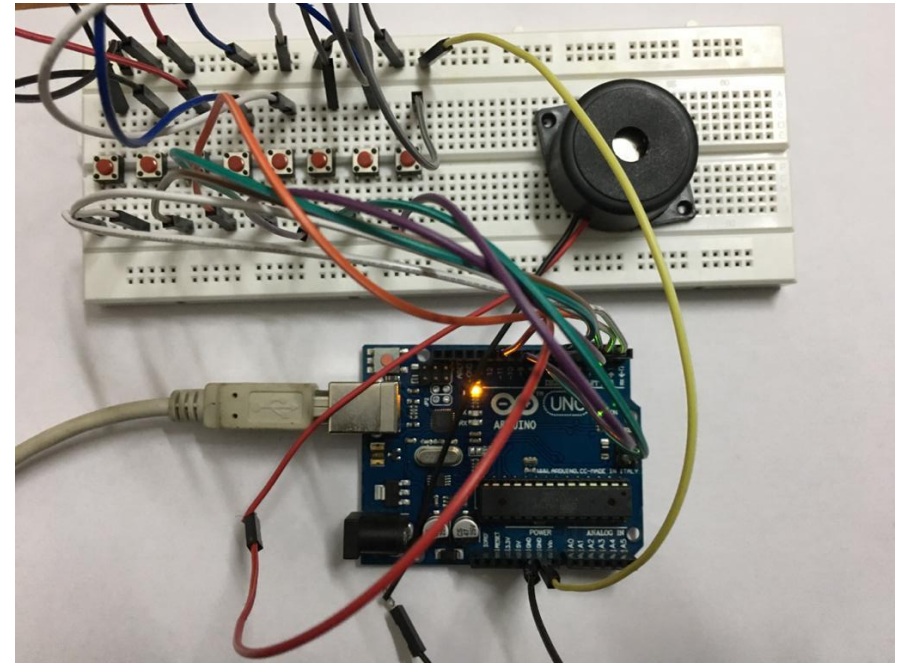
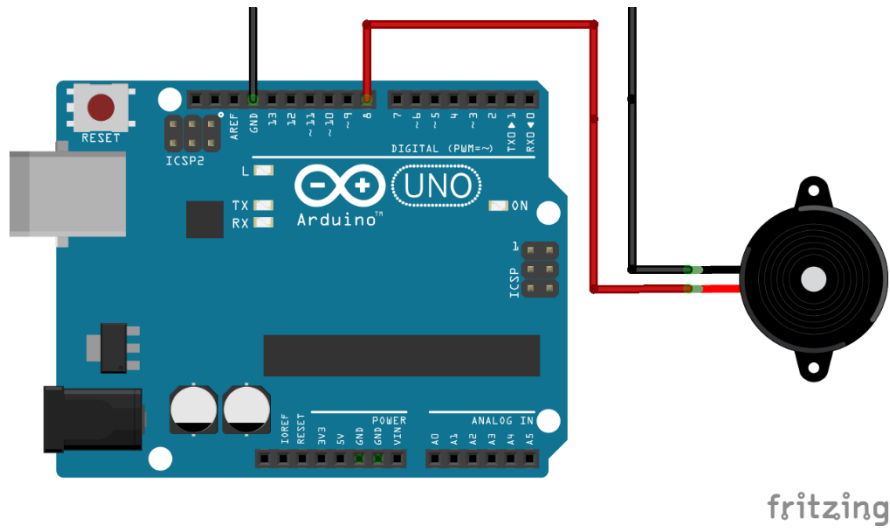


# Arduino features



- Arduino examples: Arduino codes
- If then else
- function, for loop, serial communication, serial monitor





Basic musical instrument

# Piezo buzzer (Lesson 2.6)



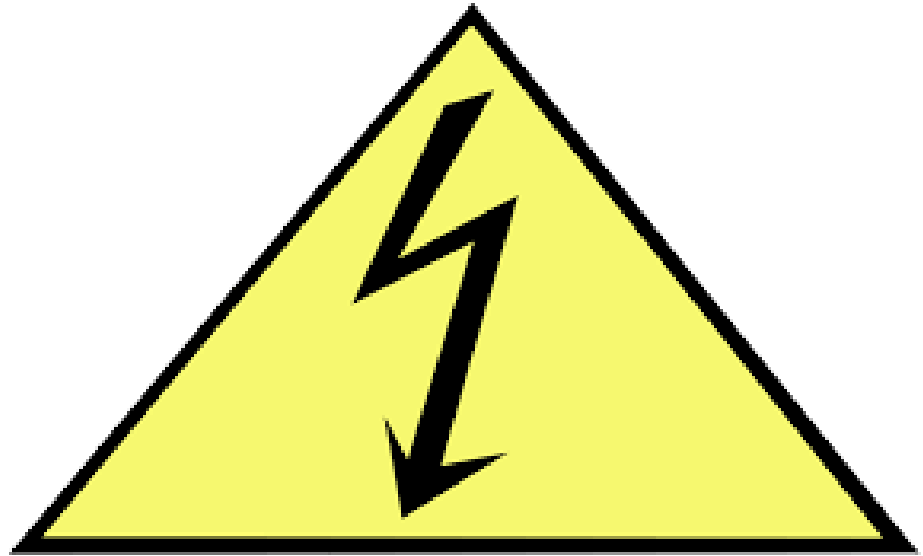
# Assignment 1



## Summary

- Algorithm, Flowchart, Programming in C, Python, Processing
- Arduino based programming
- LEDs, buttons, buzzer
- Analog and Digital signals (read and write)
- Serial communication

# Electrical Safety and handling



- [Video](#) (1:30 - 4:00 minutes)
- [Current in Arduino pins](#)
- Take utmost care of the electronics. Shouldn't be exposed to water. Keep in Ziploc bags. Delicate stuff.

# Announcements



- Assignments are skills learnt in the class
- We would give you a chance to resubmit your assignment for re-evaluation once after we evaluate your assignment.