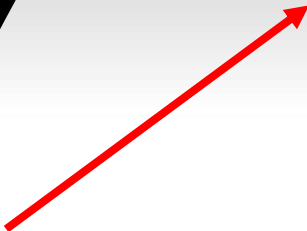




STEAM CLOWN™ PRODUCTIONS

ARDUINO STEAM ACADEMY



Art without Engineering is dreaming. Engineering without Art is calculating.

- Steven K. Roberts



STEAM CLOWN™ PRODUCTIONS

**Attribution-NonCommercial-ShareAlike
3.0 Unported (CC BY-NC-SA 3.0)**

SEE APPENDIX A, FOR LICENSING & ATTRIBUTION INFORMATION

by-nc-sa-3.0

<https://creativecommons.org/licenses/by-nc-sa/3.0/>

<https://creativecommons.org/faq/#what-does-some-rights-reserved-mean>





STEAM CLOWN™ PRODUCTIONS

CONTROL STRUCTURES



STEAM CLOWN™
& **Squeaky Hinge**
PRODUCTIONS

© Copyright 2017 STEAM Clown™

PROGRAM STRUCTURE AND CONTROL

- Program Structure
 - Define Variables
 - Setup
 - Loop
- Controlling Program Flow
 - **if**
 - **if...else**
 - **for**
 - switch case
 - **while**
 - **do... while**
 - break
 - continue
 - return
 - goto



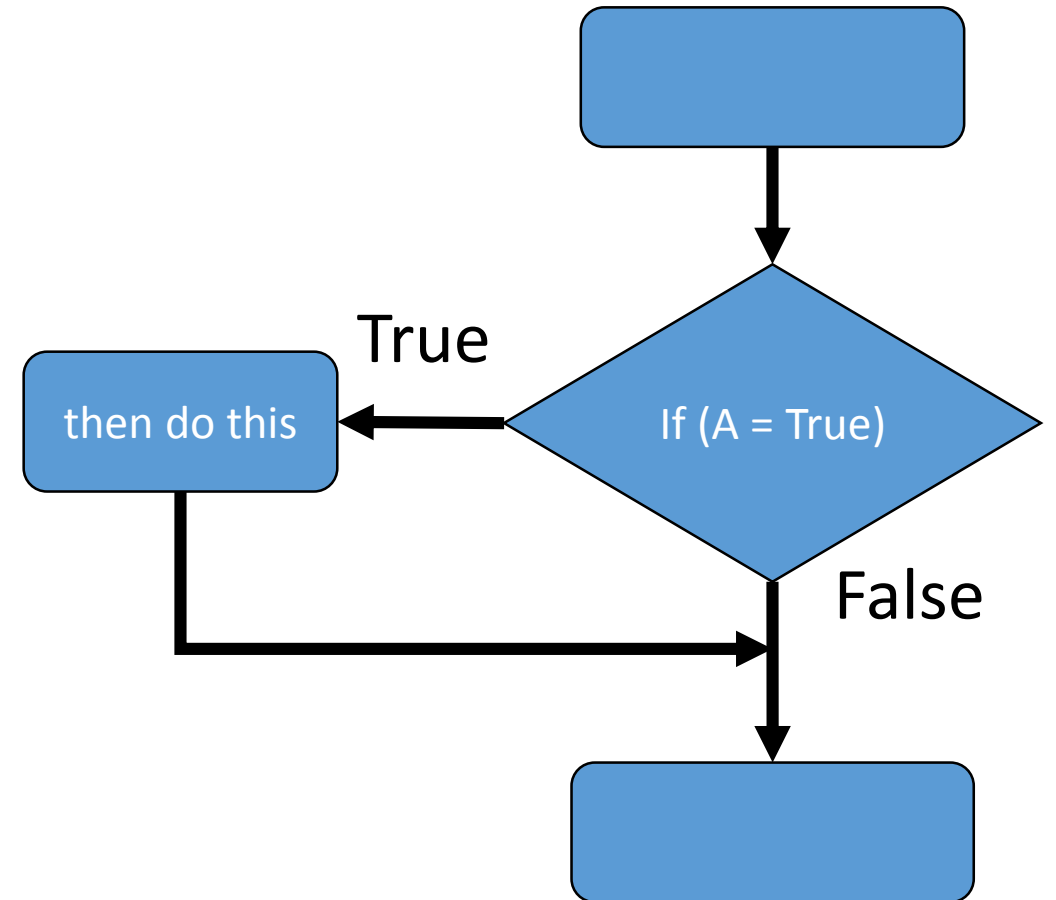
IF STATEMENT

```
If (Statement Evaluated As True)  
{  
  //do something here  
}
```

```
If (someVariable == 50)  
{  
  //do something here  
}
```

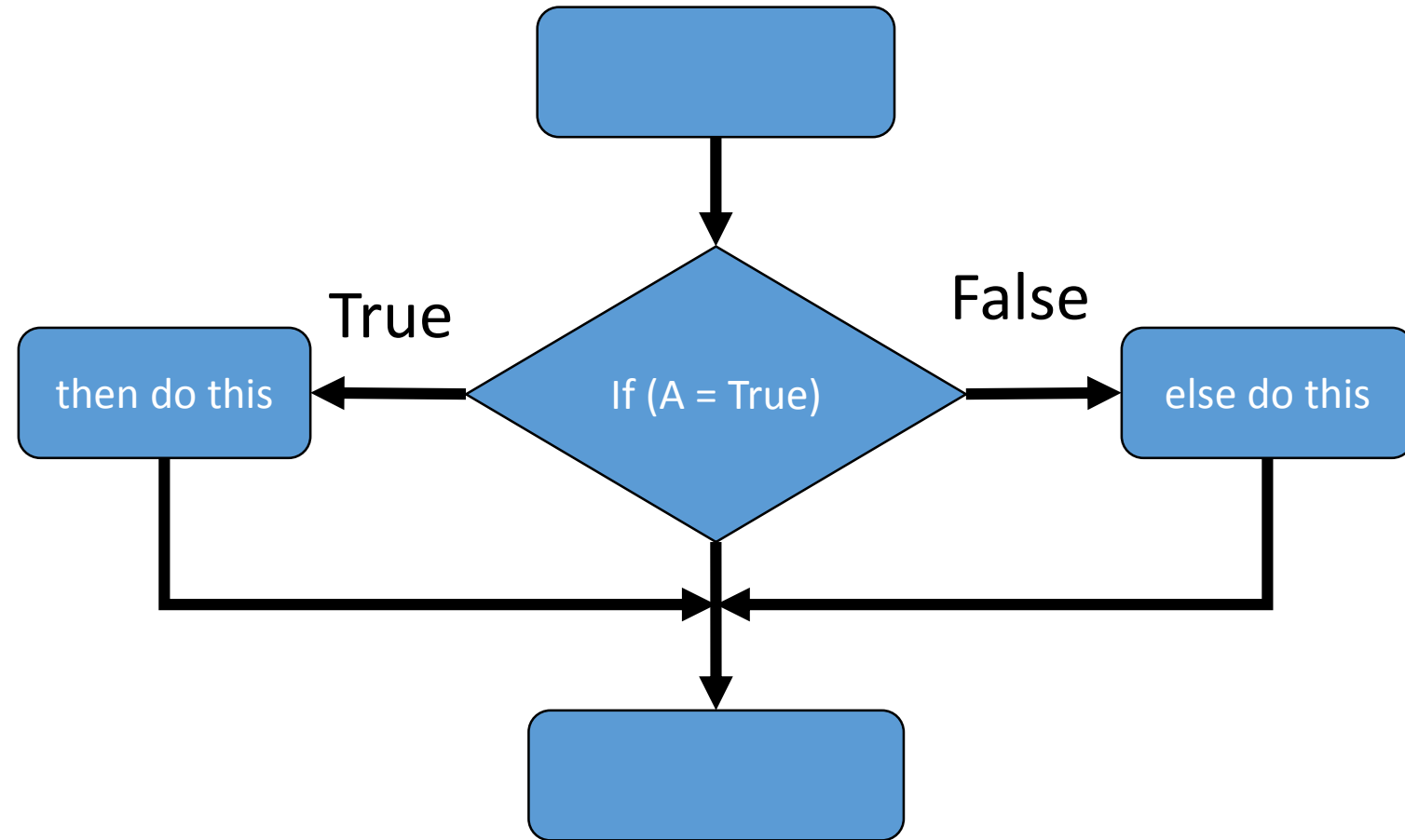
```
If (someVariable > 50)  
{  
  //do something here  
}
```

```
If (someVariable == anotherVariable)  
{  
  //do something here  
}
```



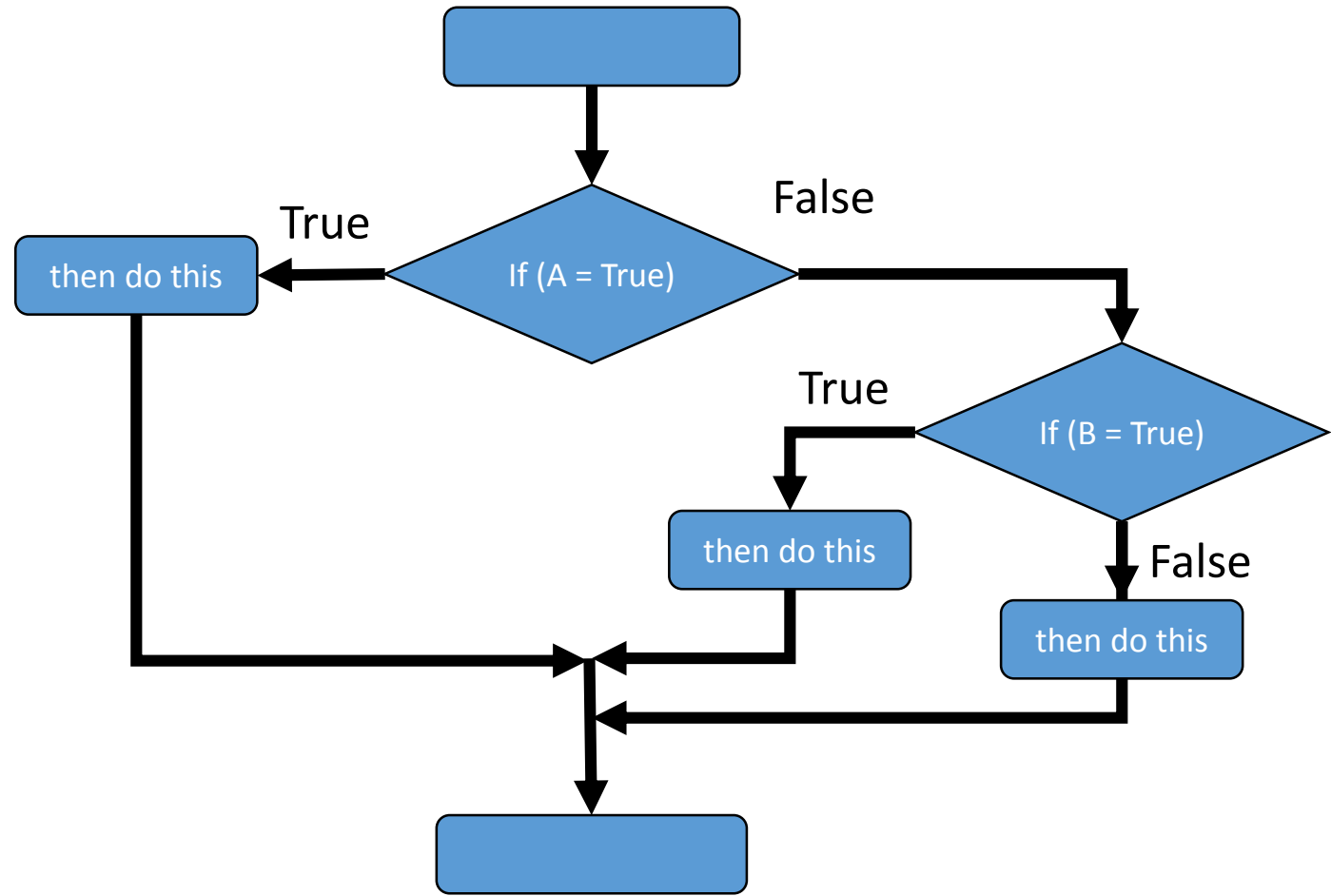
IF ELSE STATEMENT

```
if (someVariable < 500)
{
  // action A
}
else
{
  // action B
}
```

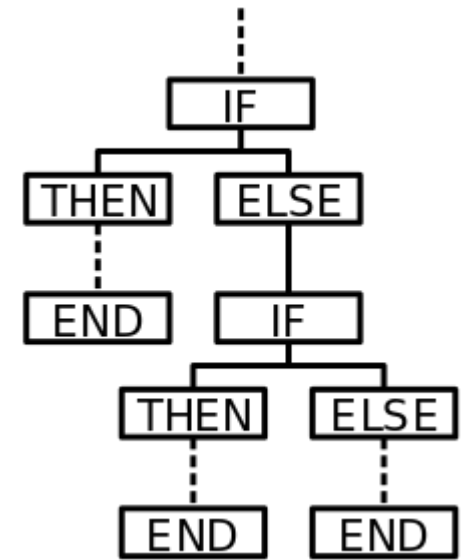
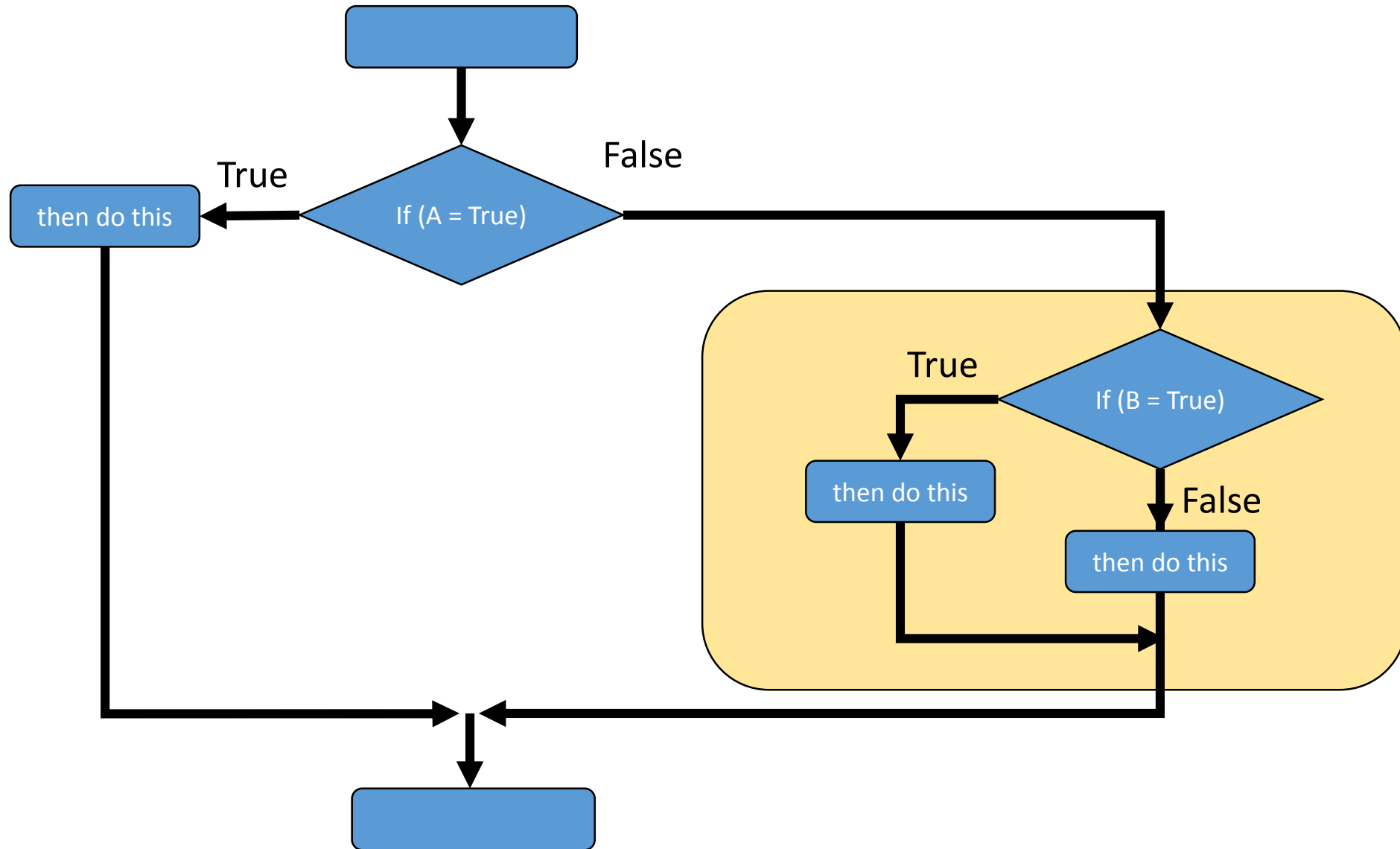


IF ELSE STATEMENT

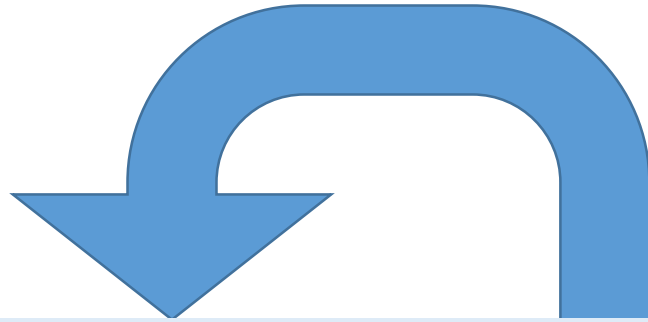
```
if (someVariable < 500)
{
  // do Thing A
}
else if (someVariable >= 1000)
{
  // do Thing B
}
else
{
  // do Thing C
}
```



NESTED IF ELSE STATEMENT



WHAT CAN BE IN A CONDITIONAL STATEMENTS



```
if (endOfWorld == TRUE && zombies > 1000000)
{
    // write code to find stuff// ...
}
```

```
if (digitalRead(2) == HIGH && digitalRead(3) == HIGH)
{
    // read two switches // ...
}
```

If the conditional statement resolves to “TRUE”



IF STATEMENT

```
If (someVariable >50)
{
//do something here
}
```

- Search and clone sketch "[steamClass IF](#)"
- What does this sketch do?
- Comparison Operators
- Lets Change it...
 - How could we change it?
 - Look back at the "Debug" Sketch

```
digitalWrite(arduinoBoardLED, HIGH);
delay(myDelayTime);
digitalWrite(arduinoBoardLED, LOW);
delay(myDelayTime);
Serial.print(".");
myDelayTime = myDelayTime + 20;
  if (myDelayTime > 500)
  {
    myDelayTime = 10;
    Serial.println("R");
  }
```

```
x == y (x is equal to y)
x != y (x is not equal to y)
x < y (x is less than y)
x > y (x is greater than y)
x <= y (x is less than or equal to y)
x >= y (x is greater than or equal to y)
```





Search the Arduino Website

Home Buy Software Products Learning Forum Support Blog



Reference Language Libraries Comparison Changes

Language Reference

Arduino programs can be divided in three main parts: *structure*, *values* (variables and constants), and *functions*.

Structure

- [setup\(\)](#)
- [loop\(\)](#)

Control Structures

- [if](#)
- [if...else](#)
- [for](#)
- [switch case](#)
- [while](#)
- [do... while](#)
- [break](#)

Variables

Constants

- [HIGH](#) | [LOW](#)
- [INPUT](#) | [OUTPUT](#) | [INPUT_PULLUP](#)
- [LED_BUILTIN](#)
- [true](#) | [false](#)
- [integer constants](#)
- [floating point constants](#)

Data Types

- [void](#)
- [boolean](#)

Functions

Digital I/O

- [pinMode\(\)](#)
- [digitalWrite\(\)](#)
- [digitalRead\(\)](#)

Analog I/O

- [analogReference\(\)](#)
- [analogRead\(\)](#)
- [analogWrite\(\)](#) - *PWM*

Due & Zero only



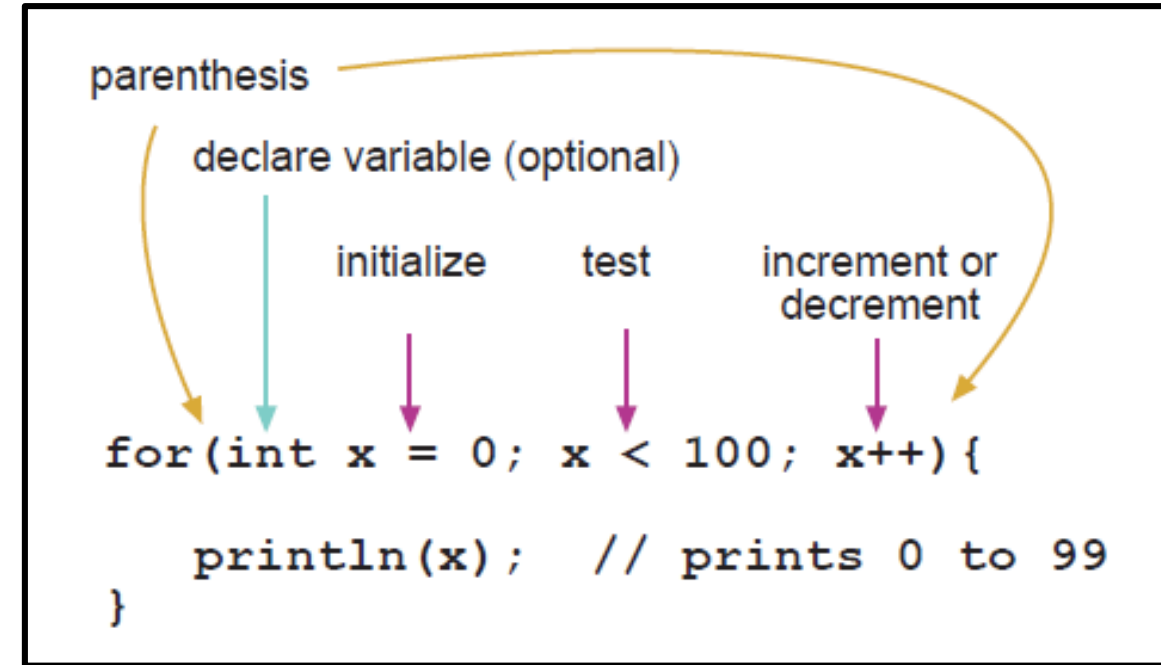
LETS DO SOME CODING - WRITE YOUR OWN SKETCH

- Review the code in the sketches and code you have done
 - Look at my sketches on [github](#) or other websites
 - Look at coding examples on [www.Arduino.Cc](#)
- Use “if”, “if/then”, “if/then/else” statements
- Make the LED blink differently based on your program control
- Before you start coding... plan it out in your lab book
- Ask me how to turn it it....



FOR LOOP STATEMENT

- Loop “for” some time...
- What does this code do?
- Why would you use code like this?



```
for (int i = 0; i < 20; i ++)  
{  
    digitalWrite(ledPin, HIGH);  
    delay(delayPeriod);  
    digitalWrite(ledPin, LOW);  
    delay(delayPeriod);  
}
```

```
for(int x = 2; x < 100; x = x * 1.5)  
{  
    println(x);  
}
```



WHILE AND DO WHILE LOOP STATEMENT

- Do Something “While” statement is “TRUE”

```
while(expression)
{
    // statement(s)
}
```

```
do
{
    // statement(s)
} while(expression)
```

```
var = 0;
while(var < 200)
{
    // do something repetitive 200 times
    var++;
}
```

```
var = 0;
do
{
    // do something repetitive 201 times
    var++;
} while(var < 200)
```

- What if we set var = 1;



WHILE AND DO WHILE LOOP

- Search and Clone
 - [steamClass whileLoop](#)
- What are these loops doing?
- Are they different?

```
loopCounter = 1;
// while(loopCounter != 10)
while(loopCounter < 10)
{
  delayTime = 200;
  digitalWrite(arduinoBoardLED, HIGH);
  delay(delayTime);
  digitalWrite(arduinoBoardLED, LOW);
  delay(delayTime);
  loopCounter++;
}
```

```
do
{
  delayTime = 500;
  digitalWrite(arduinoBoardLED, HIGH);
  delay(delayTime);
  digitalWrite(arduinoBoardLED, LOW);
  delay(delayTime);
  loopCounter++;
}while(loopCounter < 10);
```



LETS DO SOME CODING - ADD FOR AND WHILE LOOP

- Review the code in the sketches and code you have done
 - Look at my sketches on [github](#) or other websites
 - Look at coding examples on www.Arduino.Cc
- Create A New Sketch
- Add a “for” Loop and “While” Loop Statements
- Make The LED Blink Differently Based on Your Program Control
- Before you start coding... plan it out in your lab book
- Ask me how to turn it it....





STEAM CLOWN™ PRODUCTIONS

STOP HERE...



STEAM CLOWN™
& **Squeaky Hinge**
PRODUCTIONS

© Copyright 2017 STEAM Clown™



STEAM CLOWN™ PRODUCTIONS

APPENDIX



APPENDIX A: LICENSE & ATTRIBUTION

- This content is primarily the Intellectual Property of Jim Burnham, Top STEAM Clown, at STEAMClown.org
- This presentation and content is distributed under the Creative Commons License CC-by-nc-sa-3.0
- My best attempt to properly attribute, or reference any other sources or work I have used are listed in Appendix B



Under the following terms:



Attribution — You must give [appropriate credit](#), provide a link to the license, and [indicate if changes were made](#). You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.



NonCommercial — You may not use the material for [commercial purposes](#).



ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the [same license](#) as the original.

No additional restrictions — You may not apply legal terms or [technological measures](#) that legally restrict others from doing anything the license permits.



APPENDIX B: ATTRIBUTION FOR SOURCES USED



STEAM CLOWN™
& **Squeaky Hinge**
PRODUCTIONS

© Copyright 2017 STEAM Clown™



STEAM CLOWN™ PRODUCTIONS

REFERENCE SLIDES



STEAM CLOWN™
& **Squeaky Hinge**
PRODUCTIONS

© Copyright 2017 STEAM Clown™

IS IT POSSIBLE TO GET HELP?

- Git Hub – See [Steam Clown's Files](#)
- <http://www.arduino.cc/> ← Official Arduino Site
- <http://www.arduinobook.com/>
- Google Is Your Friend...
 - Google [Arduino Getting Started](#)
 - Google [Arduino Tutorials](#)
 - Google [Arduino Sketches](#)
- PDF books
 - [Arduino Programmers Notebook](#)
 - [Arduino in a Nutshell](#)
 - [Introduction to Arduino - A piece of cake!](#)
- YouTube
 - [Arduino: Your First Arduino Sketch](#)
 - [Tutorial 01 for Arduino: Getting Acquainted with Arduino](#)





STEAM CLOWN™
& **Squeaky Hinge**
PRODUCTIONS

© Copyright 2017 STEAM Clown™