FREE

IEEE - CSULB Branch
April 9 2021
Kiyo Terao
John Abrahem

Agenda

- Project
- Software setup
  - Arduino IDE
- Arduino Programming Language
  - Functions
  - Values
  - Structures

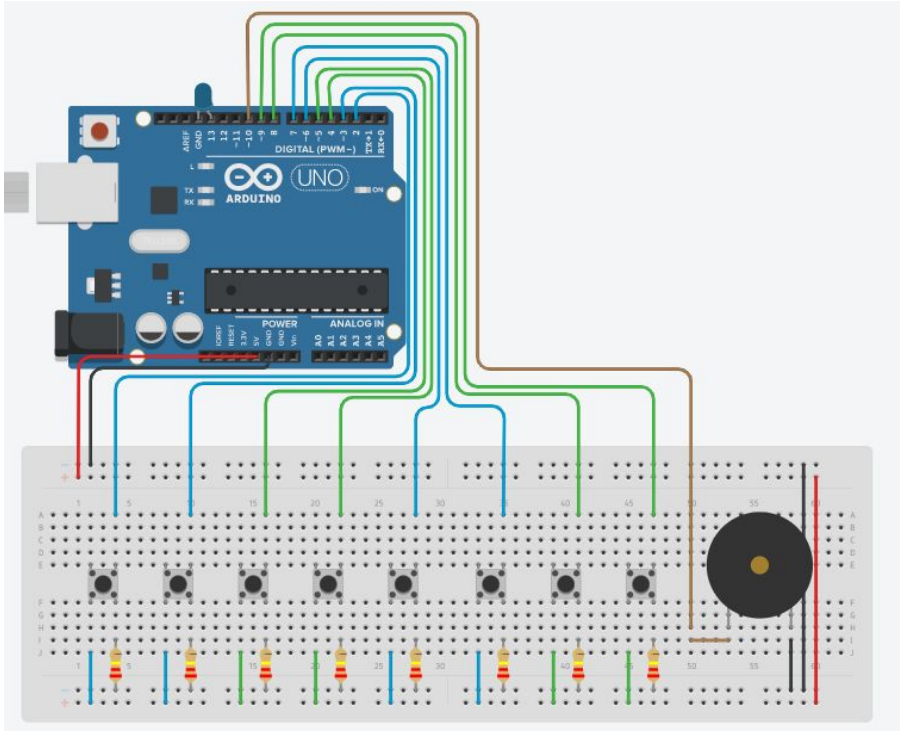
FREE

Code Link:

IEEE

Project- Arduino Piano

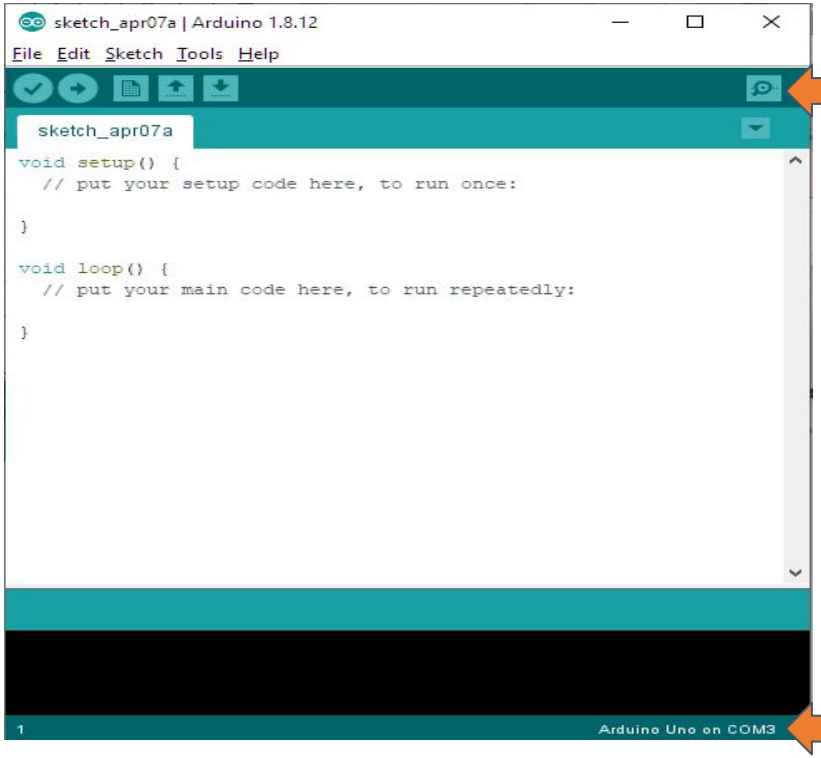
- Sound Reactive Arduino Floor Piano
- Modules used
  - Arduino Uno/nano
- Components
  - Resistors
  - Jumper wires
  - Piezo



Software Setup

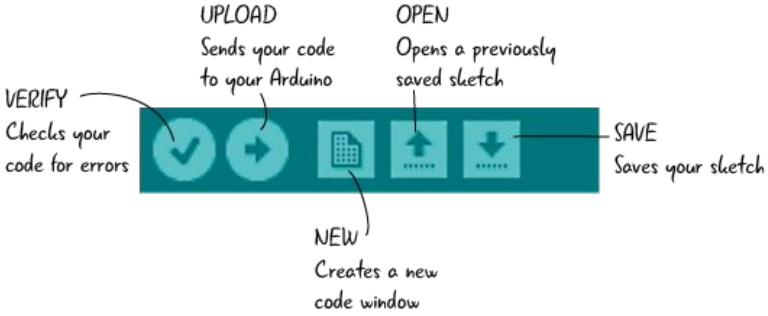
- Google “Arduino IDE”
  - Click on the first link (Software | Arduino)
  - Scroll down and look for “Arduino IDE 1.8.13 “
  - Click on your respective Operating System (i.e. Windows, Apple) to download file.

Software Setup - continued



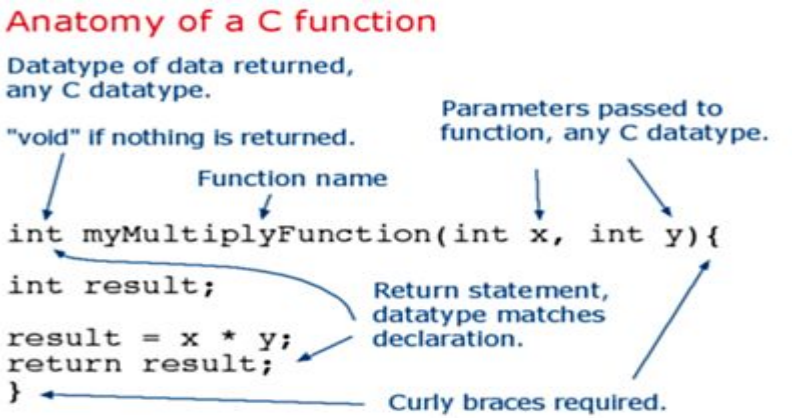
Serial Monitor

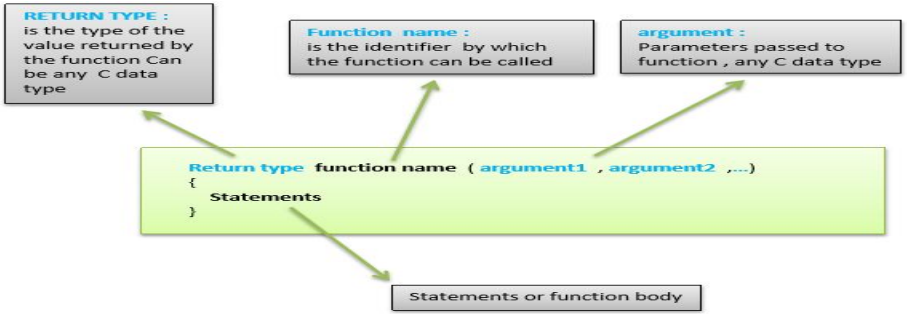
Arduino board



Arduino Programing Language -Functions

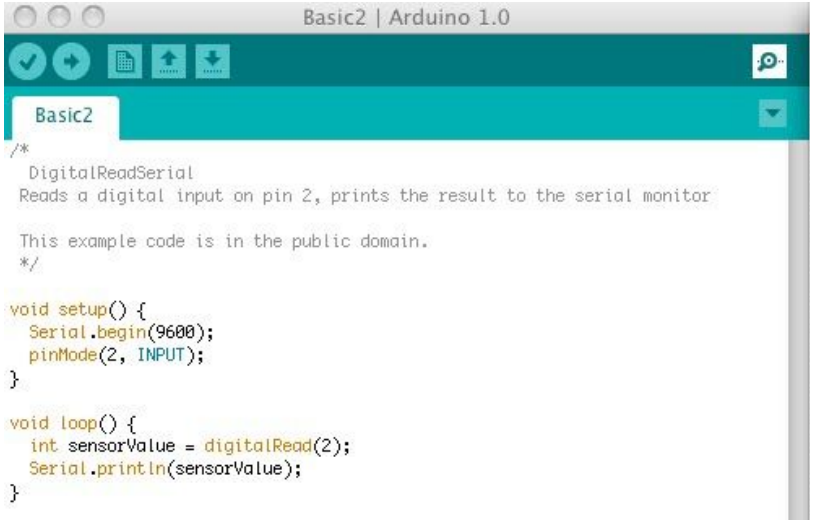
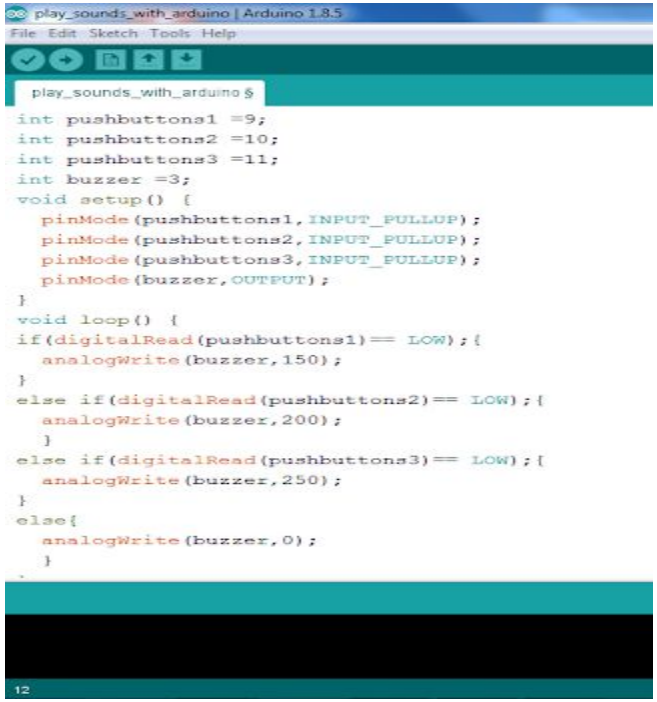
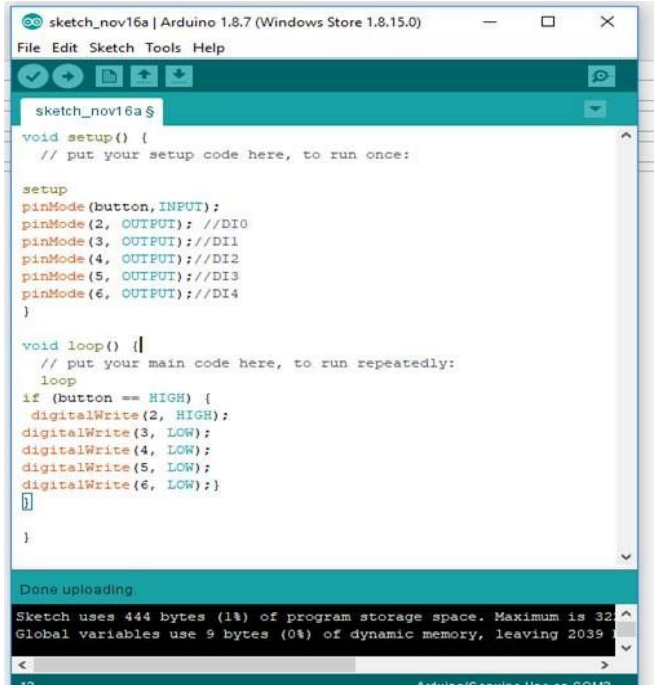
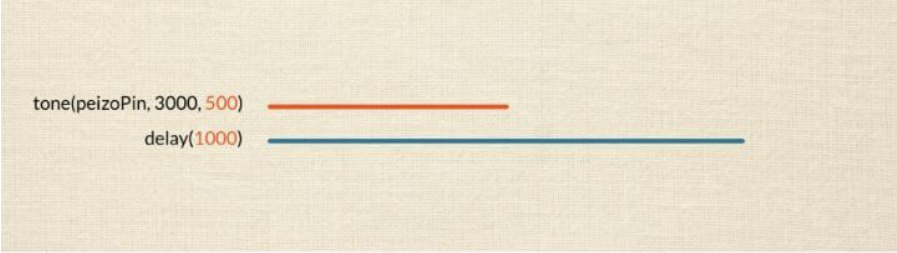
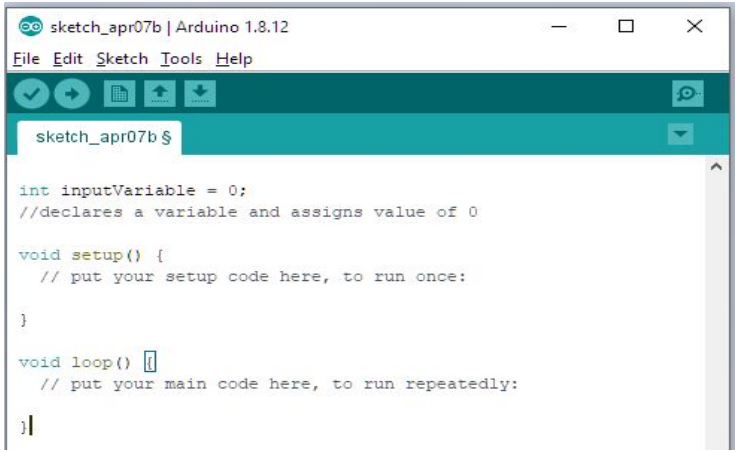
- Functions
  - Controls arduino board and performs multiple computations/tasks
  - Help the programmers stay organized
- Common functions
  - `pinMode()`
  - `digitalRead()`
  - `digitalWrite()`
- Advanced
  - `tone()`





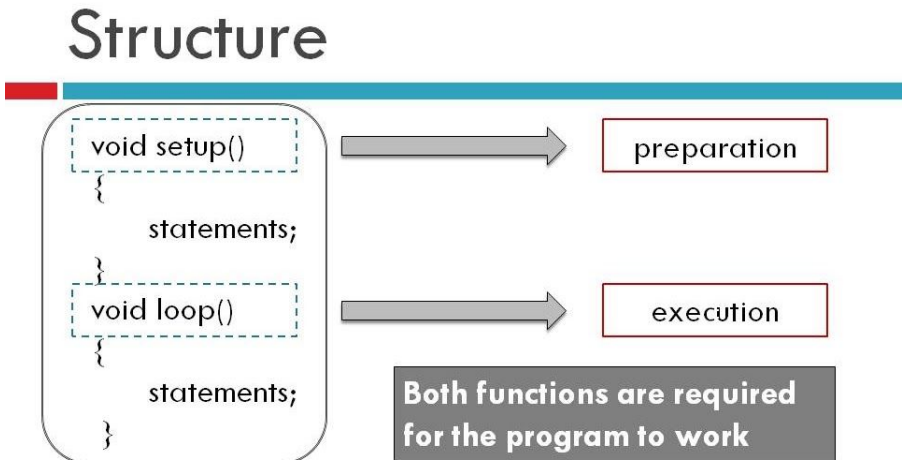
Functions > Digital I/O > pinMode()

- What is “`pinMode()`“
  - Configure the pins to act as an input or an output
  - Inside of the “void setup()”
- Syntax
  - pinMode(pin,mode)
  - Pin
- Parameters
  - pin:arduino pin number
- Example
  - ```
    void setup() {
      pinMode(cNote,INPUT);    //
    sets the digital pin 2 = cNote as
    input
    }
    ```


    ```

Functions > Digital I/O > digitalRead()

- What is “`digitalRead()`”
  - Reads the value from a specified digital pin
    - HIGH or LOW
  - Inside of the loop
- Syntax
  - digitalRead(pin)
- Parameter
  - pin : pin number you want to read
- Example
  -
```
void loop() {
  val = digitalRead(inPin);   // read the input pin
  digitalWrite(ledPin, val);  // sets the LED to the button's value
}
```



Function > digital I/O > digitalWrite()

- What is “`digitalWrite()`”
  - Used to write a HIGH or a LOW value
  - If OUTPUT with pinMode()
    - Voltage will be set to 5V/3.3V for HIGH and 0V for LOW
- Syntax
  - `digitalWrite(pin,value)` //value HIGH/LOW
- Parameter
  -
- Example
  -
```
void loop() {
  val = digitalRead(inPin);   // read the input pin
  digitalWrite(ledPin, val);  // sets the LED to the button's value
}
```



Example 2

Function > Advance I/O > tone()

- What is tone()
  - Generates a square wave of the specified frequency on a pin
- Syntax
  - `tone(pin, frequency, duration)`
- Parameters
  - frequency of the tone in hertz
  - pin: to generate the tone
- Example
  - `tone(Piezo, c, 250);  // plays note c`

    `digitalWrite(LED,HIGH);`




```
16
17  tone(peizoPin, 3000, 500);
18
19  delay(1000);
20
21 }
```

Arduino Programing Language - Variables(values & const)

- Variables (values & constants)
  - Way of naming and storing numerical values
  - A variable needs to be declared Data types and constants

ArithmaticOperator | Arduino 1.8.1

File Edit Sketch Tools Help

ArithmaticOperator

```
int Num1 = 15;
int Num2 = 3;
int Addition = Num1 + Num2;
int Subtraction = Num1 - Num2;
int Multiplication = Num1 * Num2;
int Division = Num1 / Num2;
int Remainder = Num1 % Num2;
```

Variable > Data type > int

- What is “`int`”
  - Integers are primary datatype for storing numbers
  - Whole numbers
- Syntax
  - `Int var = val`
- `Parameters`
  - `Var: variable name`
  - `Val: value assigned to the var`
- Example code
  - `int cNote = 264; //frequency of key notes`
    `//declaring the value of the keynote`



Example 2

Variable > Data types > char

- What is char
  - Used to store character
    - Stored as numbers
  - Written in single quotes
    - ‘A’
- Syntax
  - `Char var = val;`
- Parameter
  - `Var:variable name`
  - `Val:value to assign variable`
- Example
  - ```
    char myChar = 'A';
    char myChar = 65; // both are equivalent
    ```
    ```

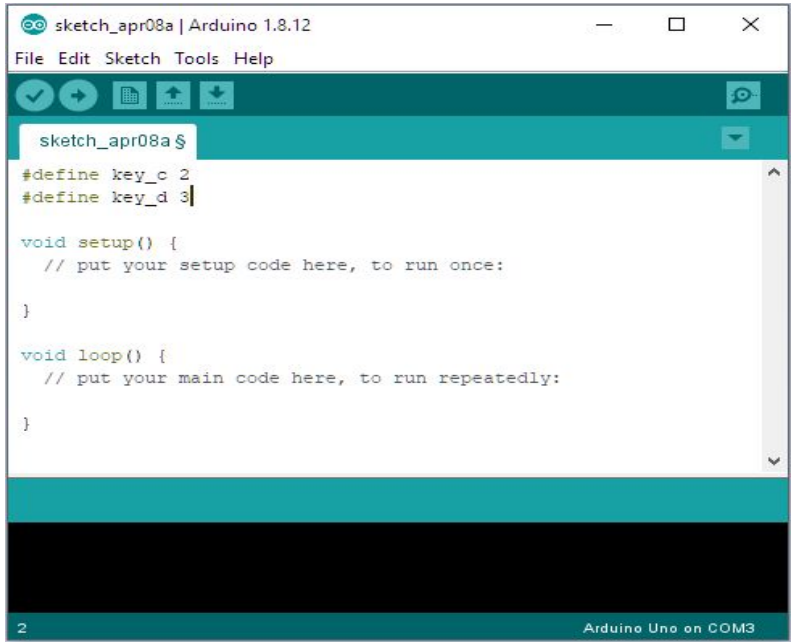
Arduino Programing Language - Structure

- Structure
  - Elements of Arduino(C++) code
  - Allows us to combine data of different types together
  - Helps to construct a complex data type
  - Can store any type of data



Structure > Syntax > #define

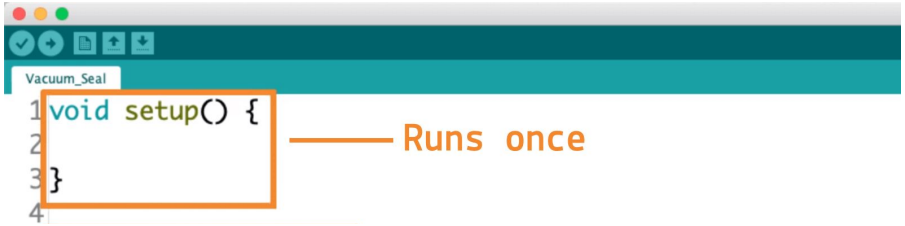
- `#define`
  - Allows the programmer to give a name to a constant value
- Syntax
  - `#define constantName  value`
- Parameter
  - constantName:name of the macro to define
  - Value: value of the macro
- Example
  - `#define cNote 2`
    `// The compiler will replace any mention of cNote with the value 2 at compile time.`



Example

Structure > Sketch > setup()

- What is “`setup()`”
  - Used for initializing
    - Variables
    - pinModes
    - libraries
  - Only run once
- Example
  -
```
void setup() {  //initializing values
  Serial.begin(9600);
  pinMode(buttonPin, INPUT);
}
```



Example 2

Structure > Sketch > loop()

- What is "`loop()`"
  - A programming structure
    - Repeats a sequence of instruction until a specific condition is met
    - Allow your program to change and respond
    - Controls arduino board
- Example code (continued from "setup()")
  -
```
void loop() {
  if (digitalRead(buttonPin) == HIGH)
{
    Serial.write('H');
  }
  else {
    Serial.write('L');
  }
  delay(1000); //milliseconds
}
```

```
5 void loop() {
6
7   //Do first...
8   //Do this next...
9   //Do this too...
10
11 }
```

Runs over and over and over...

Vacuum_Seal §

```
1 void setup() {
2
3   /* Start Serial Communication*/
4   Serial.begin(9600);
5
6 }
7
8 void loop() {
9
10  Serial.println("Ice Ice Baby");
11
12 }
```

```
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
Ice Ice Baby
```

Example 2

Structure > comparison operators > equalto

- What is ==
  - Comparison of one variable or constant against another (left and right)
- Syntax
  - `if x == y; // is true if x is equal to y and it is false if x is not equal to y`
- Parameter
  - `X:variable(int,float,double,byte,short,long)`
  - `y:variable or constant`
- Example
  -
```
if (x == y) { // tests if x is equal to y
  // do something only if the comparison
result is true
}
```

```
int a=1;

int b=1,count=0;

if(a==b) //true

count++; //count increases
```