

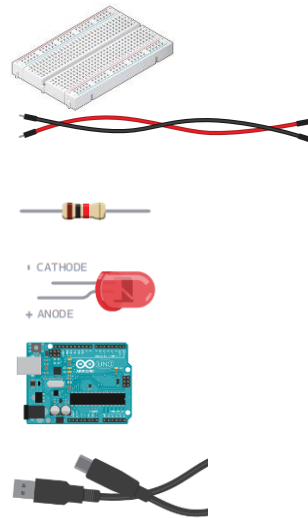
Task 1 : Build a LED Circuit on a Breadboard

Difficulty Level: Easy (Students' Expressions) (Difficulty level (1-5): 1-2)

Grade Level: 5-6

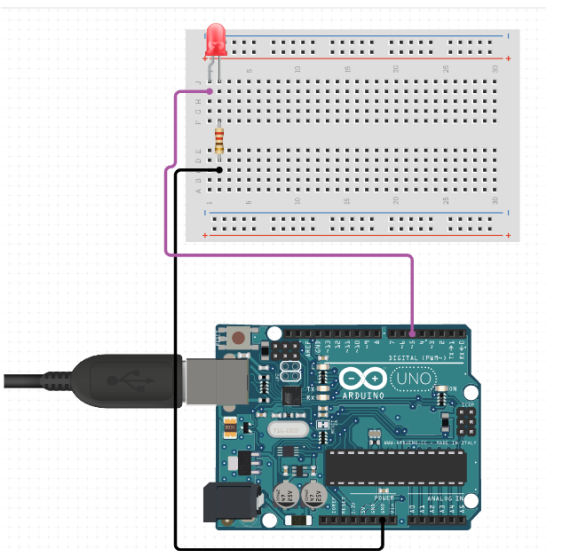
Materials :

- Breadboard
- 2 jumper wires
- One 10k ohm (brown-black-orange) resistor
- One LED
- Arduino
- USB 2.0 Cable

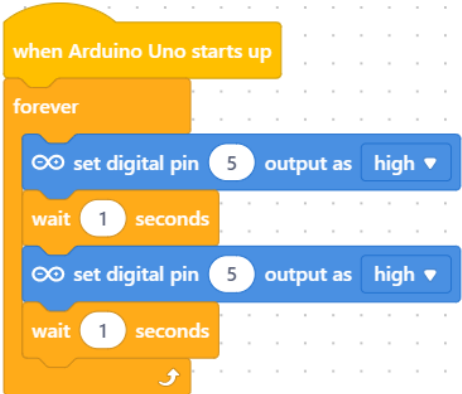


For Software : <https://ide.mblock.cc/>

Building The Circuit:

	<ol style="list-style-type: none">1. If your Arduino is connected to a battery or computer via USB, unplug it before building the circuit!2. In this circuit we see that the anode of the LED is connected via a resistor to 'pin 5' on the Arduino.3. Then the cathode of the LED is connected to the ground (GND) on the Arduino.4. After completing our circuit and writing the codes, our led connected to pin 5 will blink at 1 second intervals.
---	---

Codes:

Mlink (mBlock)	Arduino
	<pre> void setup() { pinMode(5,OUTPUT); //With this code, we choose which pin to output from //We output from pin number 5. } void loop() { digitalWrite(5,HIGH); // We make electricity pass from pin number 5, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(5,LOW); // It does not allow electricity to pass from pin 5, turns off the led. delay(1000); // We hold for 1 second. Delay(1000) value is 1 second. } </pre>

Check List:

		Yes	No	Partly
1	Did the LED connect to the digital pin (5) of the Arduino?			
2	Did a resistor been used?			
3	Is the direction of the led installed correctly?			
4	Is the other end of the led connected to GND?			
5	Is the code that runs the circuit written correctly?			
6	Is the connection between the Arduino and the software made?			
7	Uploaded the codes to Arduino?			
8	Did the blinking led circuit work?			

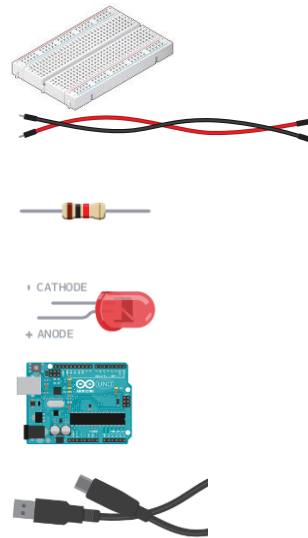
Task 2 : Build a Walking 4 LED lights

Difficulty Level: Intermediate (Students' Expressions) 3

Grade Level: 5-6

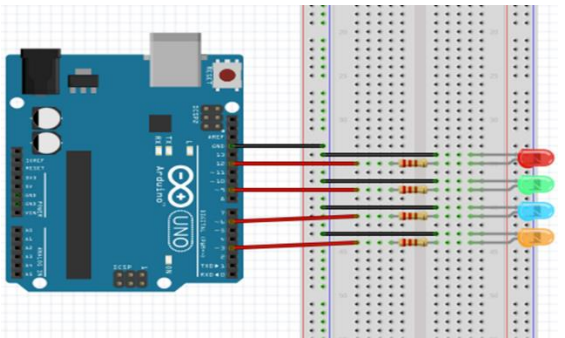
Materials :

- Breadboard
- 9 jumper wires
- 4 10k ohm (brown-black-orange) resistor
- 4 LED
- Arduino
- USB 2.0 Cable



For Software : <https://ide.mblock.cc/>

Building The Circuit:

	<ol style="list-style-type: none">1. If your Arduino is connected to a battery or computer via USB, unplug it before building the circuit!2. LEDs' (+) anode are connected to pins 3, 6, 9, 12 via Resistors.3. The (-) cathode ends of the LEDs are connected to the negative rails of the breadboard with jumper cables. Then it is connected to GND with a jumper cable.4. After completing our circuit and writing the codes, leds connected to number 12, 9, 6 and 3 pins blink for 1 second, respectively.
---	---

Codes:

Mlink (mBlock)	Arduino
	<pre>void setup() { pinMode(3,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 3 pinMode(6,OUTPUT); // With this code, we choose which pin to output from // We output from pin 6 pinMode(9,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 9 pinMode(12,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 12 } void loop() { digitalWrite(12,HIGH); // We make electricity pass from pin number 12, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(12,LOW); // It does not allow electricity to pass from pin 12, turns off the led. digitalWrite(9,HIGH); // We make electricity pass from pin number 9, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(9,LOW); // It does not allow electricity to pass from pin 9, turns off the led. digitalWrite(6,HIGH); // We make electricity pass from pin number 6, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(6,LOW); // It does not allow electricity to pass from pin 6, turns off the led. digitalWrite(3,HIGH); // We make electricity pass from pin number 3, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(3,LOW); // It does not allow electricity to pass from pin 3, turns off the led. delay(1000); // We hold for 1 second. 1000 is 1 second. }</pre>

Check List:

		Yes	No	Partly
1	Did the LEDs connect to the digital pins (12,9,6,3) of the Arduino?			
2	Did resistors been used?			
3	Is the direction of the leds installed correctly?			
4	are the other ends of the led connected to GND?			
5	Is the code that runs the circuit written correctly?			
6	Is the connection between the Arduino and the software made?			
7	Uploaded the codes to Arduino?			
8	Did the 4 blinking leds circuit work?			

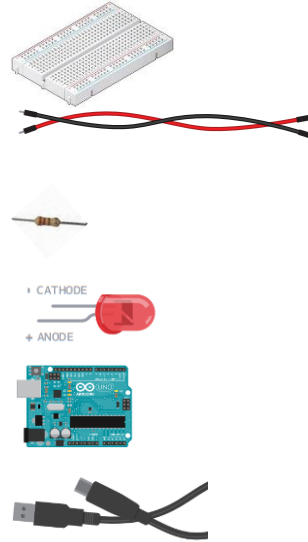
Task 3 : Build a Walking 4 parallel LED lights with single resistor (Additional task for those who have finished)

Difficulty Level: Hard (Students' Expressions) 4-5

Grade Level: 5-6

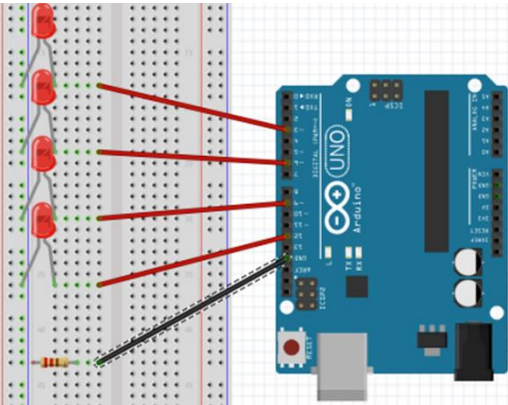
Materials :

- Breadboard
- 5 jumper wires
- One 220 ohm (red-red-brown-gold) resistor
- 4 LED
- Arduino
- USB 2.0 Cable



For Software : <https://ide.mblock.cc/>

Building The Circuit:

	<ol style="list-style-type: none">1. If your Arduino is connected to a battery or computer via USB, unplug it before building the circuit!2. The (-) cathode ends of the LEDs are connected to the negative rails of the breadboard.3. Then it is connected to GND with a resistor and jumper cable.4. LEDs' (+) anode are connected to pins 3, 6, 9, 12 via jumper cable.5. After completing our circuit and writing the codes, leds connected to number 12, 9, 6 and 3 pins, blink for 1 second, respectively.
---	--

Codes:

Mlink (mBlock)	Arduino
 <pre>when Arduino Uno starts up forever set digital pin 12 output as high wait 1 seconds set digital pin 12 output as low set digital pin 9 output as high wait 1 seconds set digital pin 9 output as low set digital pin 6 output as high wait 1 seconds set digital pin 6 output as low set digital pin 3 output as high wait 1 seconds set digital pin 3 output as low</pre>	<pre>void setup() { pinMode(3,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 3 pinMode(6,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 6 pinMode(9,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 9 pinMode(12,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 12 } void loop() { digitalWrite(12,HIGH); // We make electricity pass from pin number 12, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(12,LOW); // It does not allow electricity to pass from pin 12, turns off the led. digitalWrite(9,HIGH); // We make electricity pass from pin number 9, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(9,LOW); // It does not allow electricity to pass from pin 9, turns off the led. digitalWrite(6,HIGH); // We make electricity pass from pin number 6, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(6,LOW); // It does not allow electricity to pass from pin 6, turns off the led. digitalWrite(3,HIGH); // We make electricity pass from pin number 3, that is, the led will blink. delay(1000); // We hold for 1 second. 1000 is 1 second. digitalWrite(3,LOW); // It does not allow electricity to pass from pin 3, turns off the led. delay(1000); // We hold for 1 second. 1000 is 1 second. }</pre>

Check List:

		Yes	No	Partly
1	Did the LEDs connect to the digital pins (12,9,6,3) of the Arduino?			
2	Did one resistor be used?			
3	Is the direction of the leds installed correctly?			
4	Are the other ends of the led connected to GND?			
5	Is the code that runs the circuit written correctly via one resistor?			
6	Is the connection between the Arduino and the software made?			
7	Uploaded the codes to Arduino?			
8	Did the 4 blinking leds circuit work?			

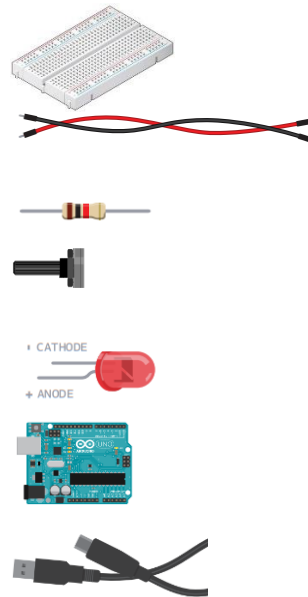
Task 4 : Build a LED Circuit with a Potentiometer to Control LED Brightness

Difficulty Level: Intermediate

Grade Level: 5-6

Materials :

- Breadboard
- 6 jumper wires
- One 10k ohm (brown-black-orange) resistor
- Potentiometer
- One LED
- Arduino
- USB 2.0 Cable

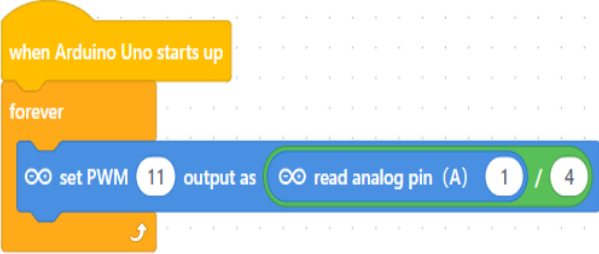


For Software : <https://ide.mblock.cc/>

Building The Circuit:

	<ol style="list-style-type: none">1. If your Arduino is connected to a battery or computer via USB, unplug it before building the circuit!2. First a black jumper wire is plugged (black is convention for GND) between a GND pin of the Arduino and the negative rails on the breadboard3. Connect the shorter leg of the LED to the ground with a small black wire.4. Plug the other (longer) leg of the LED to an independent line on the breadboard. From this leg, add a 220 Ohm resistor to yet another line.5. Add a wire between the other side of the resistor and a PWM-compatible digital pin (so we can control the brightness). Here on Arduino Uno you can choose between pins 3, 5, 6, 9, 10, and 11 – you can recognize PWM compatibility with the “~” next to the pin number.6. Plug the 3 legs of the potentiometer to 3 different lines on the breadboard.7. Connect the extreme left (or right) leg to GND.8. Connect the other extreme leg to 5V on the Arduino.9. Add a wire between the middle pin and an analog pin.
--	--

Codes:

Mlink (mBlock)	Arduino
 <p>The image shows a screenshot of the mBlock software interface. It features a yellow 'when Arduino Uno starts up' block at the top. Below it is an orange 'forever' loop block. Inside the loop, there are two blue blocks: 'set PWM 11 output as' and 'read analog pin (A) 1 / 4'. The background is a light gray grid.</p>	<pre>#define LED_PIN 11 #define POTENTIOMETER_PIN A1 // To create some defines for the pins we are going to use in the program. void setup() { pinMode(LED_PIN, OUTPUT); //The LED is a component we control, so we use pinMode() with OUTPUT.The potentiometer is a component we read from, so its mode is INPUT. However, for analog pins, no need to use pinMode() as those pins are already in INPUT mode! } void loop() { int potentiometerValue = analogRead POTENTIOMETER_PIN); //We enter the void loop(), and the first thing we do is to read the potentiometer value. To do that we use the analogRead() function, which takes one parameter: the pin number to read from. We store the value inside an integer variable. This value will be in the range 0-1023. In other words, this is a 10 bit number: 2^10 = 1024. This number corresponds to the voltage we read on the analog pin. The higher the voltage, the higher this number. For 0V you get 0, and for 5V you get 1023. int brightness = potentiometerValue / 4; analogWrite(LED_PIN, brightness); //In order to control the LED brightness, we are going to use the analogWrite() function on the LED pin (later on). This analogWrite() function takes a byte value, or in other words, a number between 0 and 255. So, before we use this function, we need to make sure the value is in the correct range. And as I really like simplicity, you can see that, roughly, 1024 is 255 multiplied by 4 (not exactly this value but this is a good enough approximate for what we need to do). So, if we want to put a value from the range 0-1023 to the range 0-255, we can just divide by 4. }</pre>

Check List:

		Yes	No	Partly
1	Did a black jumper wire is plugged between a GND pin of the Arduino and the negative rails on the breadboard?			
2	Did a resistor been used?			
3	Is the direction of the led installed correctly?			
4	Did the the shorter leg of the LED connected to the ground?			
5	Was the other (longer) leg of the LED plugged to an independent line on the breadboard and added a 220 Ohm resistor?			
6	Was a wire added between the other side of the resistor and a PWM-compatible digital pin?			
7	Was 3 legs of the potentiometer plugged to 3 different lines on the breadboar?			
8	Is the connection between the Arduino and the software made?			
9	Uploaded the codes to Arduino?			
10	Was LED brightness was controlled by potentiometer?			

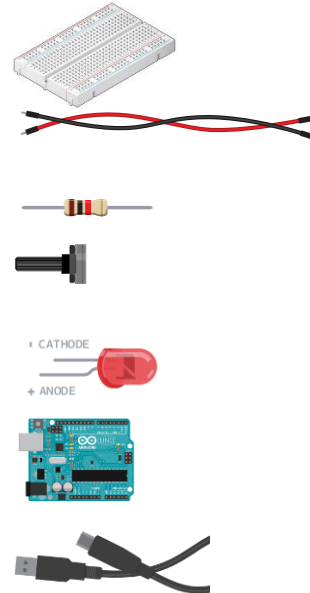
Task 5 : Creating the Circuit that Provides Speed Control of the LEDs in the Walking Light Circuit Using a Potentiometer

Difficulty Level: Intermediate

Grade Level: 5-6

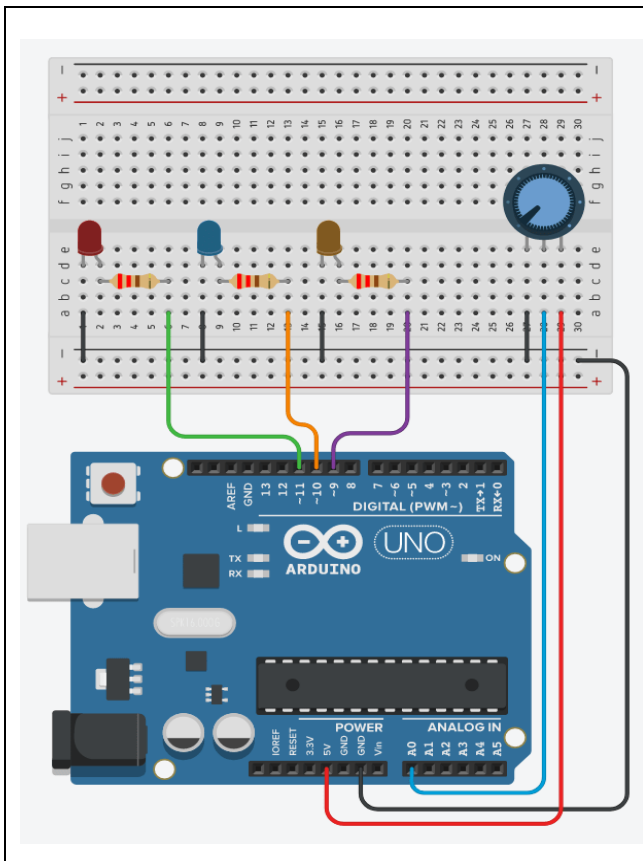
Materials :

- Breadboard
- 7 jumper wires
- 4 10k ohm (brown-black-orange) resistor
- Potentiometer
- 3 LED
- Arduino
- USB 2.0 Cable



For Software : <https://ide.mblock.cc/>

Building The Circuit:



1. If your Arduino is connected to a battery or computer via USB, unplug it before building the circuit!

2. First a black jumper wire is plugged (black is convention for GND) between a GND pin of the Arduino and the negative rails on the breadboard
3. Connect the shorter legs of the LEDs to the ground with a small black wire.

4. Plug the other (longer) legs of the LEDs to an independent line on the breadboard. From this legs, add a 220 Ohm resistor to yet another line.

5. Add a wire between the other side of the resistor and a PWM-compatible digital pin (so we can control the brightness). Here on Arduino Uno you can choose between pins 3, 5, 6, 9, 10, and 11 – you can recognize PWM compatibility with the “~” next to the pin number.

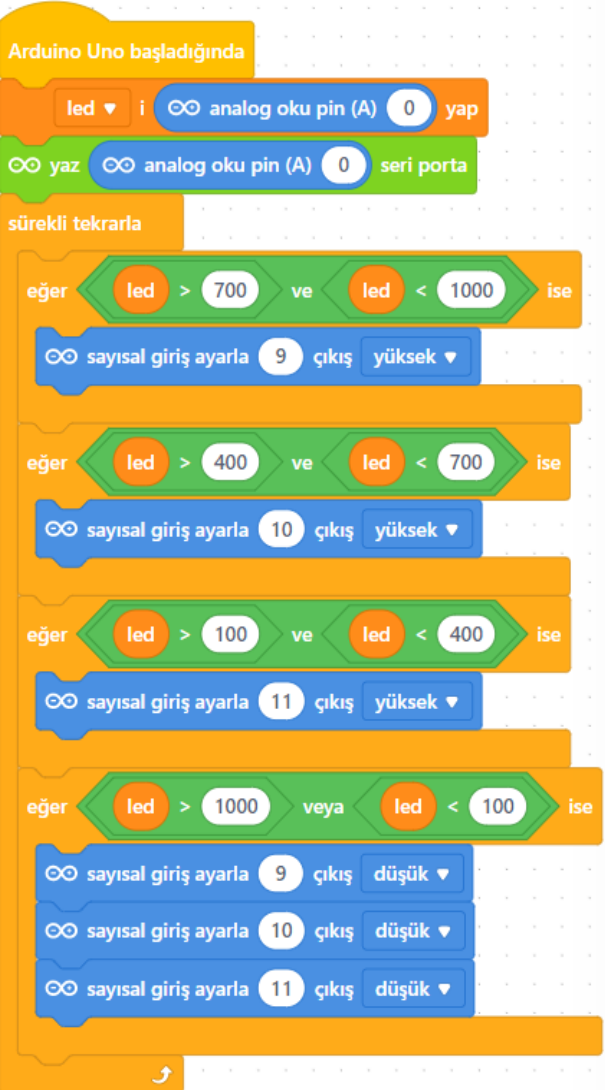
6. Plug the 3 legs of the potentiometer to 3 different lines on the breadboard.

7. Connect the extreme left (or right) leg to GND.

8. Connect the other extreme leg to 5V on the Arduino.

9. Add a wire between the middle pin and an analog pin.

Codes:

Mlink (mBlock)	Arduino
 <p>Arduino Uno başladığında</p> <p>led i analog oku pin (A) 0 yap</p> <p>yaz analog oku pin (A) 0 seri porta</p> <p>sürekli tekrarla</p> <p>eğer led > 700 ve led < 1000 ise</p> <p>sayısal giriş ayarla 9 çıkış yüksek</p> <p>eğer led > 400 ve led < 700 ise</p> <p>sayısal giriş ayarla 10 çıkış yüksek</p> <p>eğer led > 100 ve led < 400 ise</p> <p>sayısal giriş ayarla 11 çıkış yüksek</p> <p>eğer led > 1000 veya led < 100 ise</p> <p>sayısal giriş ayarla 9 çıkış düşük</p> <p>sayısal giriş ayarla 10 çıkış düşük</p> <p>sayısal giriş ayarla 11 çıkış düşük</p>	<pre>void setup() { pinMode(9,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 9 pinMode(10,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 10 pinMode(11,OUTPUT); // With this code, we choose which pin to output from. //We output from pin 11 pinMode(A0,INPUT); // With this code, we choose which pin to input from. led = analogRead(A0); } void loop() { while(1) { if((led > 700) && (led < 1000)) { digitalWrite(9,1); // We make electricity pass from pin number 9, that is, the led will blink. } if((led > 400) && (led < 700)) { digitalWrite(10,1); // We make electricity pass from pin number 10, that is, the led will blink. } if((led > 100) && (led < 400)) { digitalWrite(11,1); // We make electricity pass from pin number 11, that is, the led will blink. } if((led > 1000) (led < 100)) { digitalWrite(9,0); It does not allow electricity to pass from pin 9, turns off the led. digitalWrite(10,0); It does not allow electricity to pass from pin 10, turns off the led. digitalWrite(11,0); It does not allow electricity to pass from pin 11, turns off the led. } } }</pre>

Check List:

		Yes	No	Partly
1	Are 3 leds connected to different pins of Arduino?			
2	Is the resistor used?			
3	Is the walking light circuit installed?			
4	Have the potentiometer connected to the 5v, GND and Analog pins?			
5	Is analog and PWM conversion done?			
6	Is the connection between the Arduino and the software made?			
7	Have you made the connection of the software with the Arduino?			
8	Is the speed control code written?			
9	It did error checking. They replaced the faulty elements in the circuit. Changed the circuit setup developed			
10	Uploaded the codes to Arduino?			
11	Developed their code changed improved (He wrote his code in fewer lines)			