



Model used to generate the content: Pro Search

The link of the conversation : <https://www.perplexity.ai/search/int-ledpins-12111098-void-setup-e3WTKcz6RFa5wyJAZw6qpA#0>

```
int LedPINS []={12,11,10,9,8};

void setup() {
  int i=0;
  while(i<5){
    pinMode (LedPINS[i],OUTPUT);
    i++;
  }
}

void loop() {
  int x=0;
  while(x<5){
    int brightness=0;
    while(brightness<=255){
      analogWrite(LedPINS[x], brightness);
      brightness++;
    }
    x++;
    delay (1000);
  }
  int y=0;
  while(y<5){
    int brightness = 0;
    while(brightness<=0){
      analogWrite(LedPINS[y], brightness);
      brightness++;
    }
  }
```

```
y++;  
delay (1000);  
}  
}
```

Use pins 8 to 12 for the LEDs

- Running light from 12 to 8 with a delay of 1 second (turn all LED on one by one, then turn all LED off one by one)
- use `analogWrite()` to control the brightness of LEDs
- implement using `while()` loop and an array to set the pin modes

make them also fade

The current code has the right idea but needs fixes and improvements for the running light from pin 12 down to 8 with fading in and fading out of LEDs using `analogWrite()` and while loops with arrays.

Below is a corrected and improved Arduino code example that:

- Uses pins 12 to 8 (LEDPINS array) for LEDs
- Sets pin modes in `setup()` with while loop
- In `loop()`, runs LEDs from pin 12 to 8 sequentially
- Fades them ON with `analogWrite` from 0 to 255 brightness
- Introduces a 1-second delay after each LED fully turns ON
- Then fades LEDs OFF from 255 to 0 brightness in order
- Introduces a 1-second delay after each LED fully turns OFF
- Uses while loops for everything as requested

```
int LedPINS[] = {12, 11, 10, 9, 8};  
  
void setup() {  
  int i = 0;  
  while (i < 5) {  
    pinMode(LedPINS[i], OUTPUT);  
    i++;  
  }  
}
```

```

void loop() {
    int x = 0;

    // Fade ON from pin 12 to 8
    while (x < 5) {
        int brightness = 0;
        while (brightness <= 255) {
            analogWrite(LedPINS[x], brightness);
            delay(10); // small delay for fading effect smoothness
            brightness++;
        }
        delay(1000); // 1 second delay after each LED fully on
        x++;
    }

    int y = 0;

    // Fade OFF from pin 12 to 8
    while (y < 5) {
        int brightness = 255;
        while (brightness >= 0) {
            analogWrite(LedPINS[y], brightness);
            delay(10); // small fade delay
            brightness--;
        }
        delay(1000); // 1 second delay after each LED fully off
        y++;
    }
}

```

Explanation:

- The outer while loops go through each LED pin from 12 to 8 (index 0 to 4)
- The inner while loops increment or decrement brightness from 0 to 255 or 255 to 0 for fade effect, using `analogWrite()`
- Added short 10 ms delay inside brightness loop for visible fading instead of instantaneous jumps
- Added 1-second delay after each full fade ON or fade OFF to meet requirement

This will create a running lights effect that fades LEDs on one by one from pin 12 to 8, holds them on for 1 second, then fades them off one by one in the same order with 1-second delay, using analog brightness control. All loops use `while()` and array for pin management as requested.