

# Oficina de Programação utilizando a plataforma Arduino

Bianca Alessandra Visineski Alberton  
bi.alberton@gmail.com

Paulo Nesello Künzel  
kruger\_paulo@hotmail.com

Programa de Educação Tutorial  
Computando Culturas em Equidade  
Departamento Acadêmico de Informática  
UTFPR

06 de outubro de 2012



## O que há de novo?

## O que há de novo?

Funções da Biblioteca Arduino

## O que há de novo?

### Funções da Biblioteca Arduino

- `randomSeed(analogRead(0));` - Determina valores aleatórios a partir de um valor inicial definido pela entrada analógica.

## O que há de novo?

### Funções da Biblioteca Arduino

- *randomSeed(analogRead(0))*; - Determina valores aleatórios a partir de um valor inicial definido pela entrada analógica.
- *random(valor\_ maximo)*; - Gera um valor aleatório dentro dos valores de *randomSeed()*, ate um valor máximo definido por parâmetro.

## O que há de novo?

### Funções da Biblioteca Arduino

- *randomSeed(analogRead(0))*; - Determina valores aleatórios a partir de um valor inicial definido pela entrada analógica.
- *random(valor\_maximo)*; - Gera um valor aleatório dentro dos valores de *randomSeed()*, ate um valor máximo definido por parâmetro.
- *random(inicio,fim)*; - Gera valores aleatórios entre os valores enviados por parâmetro.

## O que há de novo?

### Funções da Biblioteca Arduino

- *randomSeed(analogRead(0))*; - Determina valores aleatórios a partir de um valor inicial definido pela entrada analógica.
- *random(valor\_maximo)*; - Gera um valor aleatório dentro dos valores de *randomSeed()*, ate um valor máximo definido por parâmetro.
- *random(inicio,fim)*; - Gera valores aleatórios entre os valores enviados por parâmetro.
- *constrain(RGB2[x], 0, 255)*; - Garante que o valor dado seja não negativo e esteja dentro da faixa escolhida.



Primeiramente, como usar as cores vermelho, verde e azul para conseguir outras cores?

**Tabela :** Cores básicas com combinação RGB.

<b>RED</b>	<b>GREEN</b>	<b>BLUE</b>	<b>Cor</b>
255	0	0	Vermelho
0	255	0	Verde
0	0	255	Azul
255	255	0	Amarelo
0	255	255	Azul Piscina (Ciano)
255	0	255	Magenta
255	255	255	Branco

# Código I

```
1 // Projeto 6 - Lidando com cores
2
3 float RGB1[3];
4 float RGB2[3];
5 float INC[3];
6 int red, green, blue;
7 int RedPin = 11;
8 int GreenPin = 10;
9 int BluePin = 9;
10
11 void setup() {
12     randomSeed(analogRead(0));
13
14     RGB1[0] = 0;
15     RGB1[1] = 0;
16     RGB1[2] = 0;
17
18     RGB2[0] = random(256);
19     RGB2[1] = random(256);
```

# Código II

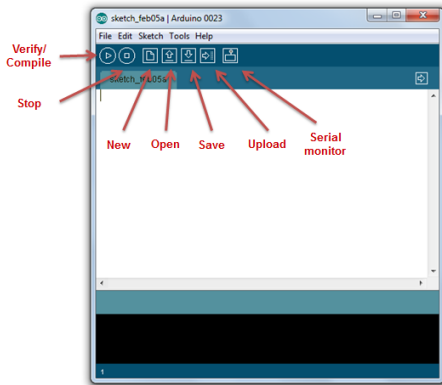
```
20     RGB2[2] = random(256);
21 }
22
23 void loop() {
24     randomSeed(analogRead(0));
25
26     for (int x=0; x<3; x++) {
27         INC[x] = (RGB1[x] - RGB2[x]) / 256;
28     }
29
30     for (int x=0; x<256; x++) {
31         red = int(RGB1[0]);
32         green = int(RGB1[1]);
33         blue = int(RGB1[2]);
34
35         analogWrite (RedPin, red);
36         analogWrite (GreenPin, green);
37         analogWrite (BluePin, blue);
38         delay(100);
```

```
39
40     RGB1[0] -= INC[0];
41     RGB1[1] -= INC[1];
42     RGB1[2] -= INC[2];
43 }
44
45 for (int x=0; x<3; x++) {
46     RGB2[x] = random(556)-300;
47     RGB2[x] = constrain(RGB2[x], 0, 255);
48     delay(1000);
49 }
50 }
```

---

# Projeto 7.1 - Luminária de humor com entrada serial

O controle agora será feito pelo Serial monitor.



**Figura :** Tela inicial.

## O que há de novo?

## O que há de novo?

### Variáveis

- `char buffer[18];` - Strings ou palavras

## O que há de novo?

### Variáveis

- `char buffer[18];` - Strings ou palavras

### Estruturas de programação

- `while(condicao)` - Loop "parente" do for



## O que há de novo?

### Variáveis

- *char buffer[18];* - Strings ou palavras

### Estruturas de programação

- *while(condicao)* - Loop "parente" do for
- *char\* parameter;* - Ponteiros

## Funções da Biblioteca Arduino

## Funções da Biblioteca Arduino

- *Serial.begin(9600);* - Inicia uma comunicação serial com o computador por meio da porta USB.

## Funções da Biblioteca Arduino

- *Serial.begin(9600);* - Inicia uma comunicação serial com o computador por meio da porta USB.
- *Serial.flush();* - "Limpa" tudo o que estiver na comunicação serial

## Funções da Biblioteca Arduino

- *Serial.begin(9600);* - Inicia uma comunicação serial com o computador por meio da porta USB.
- *Serial.flush();* - "Limpa" tudo o que estiver na comunicação serial
- *Serial.available();* - Verifica se quaisquer caracteres tenham sido enviados pela comunicação serial.

## Funções da Biblioteca Arduino

- *Serial.begin(9600);* - Inicia uma comunicação serial com o computador por meio da porta USB.
- *Serial.flush();* - "Limpa" tudo o que estiver na comunicação serial
- *Serial.available();* - Verifica se quaisquer caracteres tenham sido enviados pela comunicação serial.
- *Serial.read();* - Lê os valores da entrada serial.

## Funções da Biblioteca Arduino

- *Serial.begin(9600);* - Inicia uma comunicação serial com o computador por meio da porta USB.
- *Serial.flush();* - "Limpa" tudo o que estiver na comunicação serial
- *Serial.available();* - Verifica se quaisquer caracteres tenham sido enviados pela comunicação serial.
- *Serial.read();* - Lê os valores da entrada serial.
- *Serial.print(dados);* - Envia valores pela porta serial, que são exibidos no monitor

## Funções da Biblioteca Arduino

- *Serial.begin(9600);* - Inicia uma comunicação serial com o computador por meio da porta USB.
- *Serial.flush();* - "Limpa" tudo o que estiver na comunicação serial
- *Serial.available();* - Verifica se quaisquer caracteres tenham sido enviados pela comunicação serial.
- *Serial.read();* - Lê os valores da entrada serial.
- *Serial.print(dados);* - Envia valores pela porta serial, que são exibidos no monitor
- *Serial.println(dados);* - Idem ao *Serial.print(dados);*, porém pula linha quando chega no final do texto.



## Funções da Biblioteca Arduino

- *Serial.begin(9600);* - Inicia uma comunicação serial com o computador por meio da porta USB.
- *Serial.flush();* - "Limpa" tudo o que estiver na comunicação serial
- *Serial.available();* - Verifica se quaisquer caracteres tenham sido enviados pela comunicação serial.
- *Serial.read();* - Lê os valores da entrada serial.
- *Serial.print(dados);* - Envia valores pela porta serial, que são exibidos no monitor
- *Serial.println(dados);* - Idem ao *Serial.print(dados);*, porém pula linha quando chega no final do texto.
- *strtok (data, " ,");* - Separa uma string de acordo com termos-chave (Nesse caso é a vírgula)

# Código 1

```
1 // Luminaria RGB controlada por entrada serial
2
3 char buffer[18];
4 int red, green, blue;
5
6 int RedPin = 11;
7 int GreenPin = 10;
8 int BluePin = 9;
9
10 void setup() {
11     Serial.begin(9600);
12     Serial.flush();
13     pinMode(RedPin, OUTPUT);
14     pinMode(GreenPin, OUTPUT);
15     pinMode(BluePin, OUTPUT);
16 }
17
18 void loop() {
19     if (Serial.available() > 0) {
```

# Código II

```
20         int index=0;
21         delay(100);
22         int numChar = Serial.available();
23
24             if (numChar>15) {
25                 numChar=15;
26             }
27         while (numChar-->0) {
28             buffer[index++] = Serial.read();
29         }
30         splitString(buffer);
31     }
32 }
33
34 void splitString(char* data) {
35     Serial.print("Data entered: ");
36     Serial.println(data);
37     char* parameter;
38     parameter = strtok (data, " ,");
```

# Código III

```
39
40     while (parameter != NULL) {
41         setLED(parameter);
42         parameter = strtok (NULL, ‘ ‘ ,’);
43     }
44     for (int x=0; x<16; x++) { // limpa o buffer e o
45         texto do array buffer
46         buffer[x]='\0';
47     }
48     Serial.flush();
49 }
50
51 void setLED(char* data) {
52     if ((data[0] == 'r') || (data[0] == 'R')) {
53         int Ans = strtol(data+1, NULL, 10);
54         Ans = constrain(Ans,0,255);
55         analogWrite(RedPin, Ans);
56         Serial.print(‘ ‘Red is set to: ’’);
```

```
57     Serial.println(Ans);
58 }
59 if ((data[0] == 'g') || (data[0] == 'G')) {
60     int Ans = strtol(data+1, NULL, 10);
61     Ans = constrain(Ans,0,255);
62     analogWrite(GreenPin, Ans);
63     Serial.print("Green is set to: ");
64     Serial.println(Ans);
65 }
66 if ((data[0] == 'b') || (data[0] == 'B')) {
67     int Ans = strtol(data+1, NULL, 10);
68     Ans = constrain(Ans,0,255);
69     analogWrite(BluePin, Ans);
70     Serial.print("Blue is set to: ");
71     Serial.println(Ans);
72 }
73 }
```

## Exercício

Para um circuito com 2 LED's vermelhos e e 2 LED's verdes, faça um programa que quando o usuário digitar "VERDE" no Serial Monitor os LED's verdes ascendam e quando o usuário digitar "VERMELHO" os dois LED's ascendam.

OBS: os LED's vermelhos e verdes nunca podem estar acesos ao mesmo tempo.