

Lista de Exercício

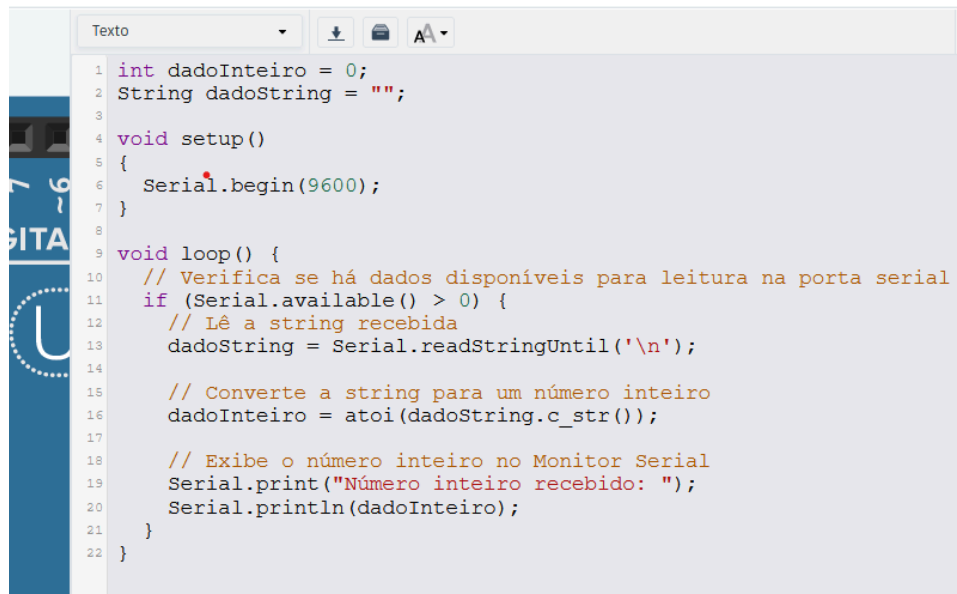
1. **Cálculo da média de três números:** Peça ao usuário para inserir três números e calcule a média deles.
2. **Fatorial de um número:** Escreva um programa que calcule o fatorial de um número fornecido pelo usuário.
3. **Contagem regressiva:** Faça um programa que solicite ao usuário um número e exiba uma contagem regressiva desse número até zero.
4. **Ordenação de números:** Crie um programa que solicite ao usuário três números e os exiba em ordem crescente.
5. Desenvolva um programa para o Arduino que leia a luminosidade utilizando um sensor LDR conectado ao pino PWM de sua escolha. O programa deve exibir a luminosidade medida no Monitor Serial a cada 5 segundos. Quando a luminosidade estiver abaixo de um determinado limite fornecido pelo usuário e o interruptor deslizante estiver na posição LIGADO, o LED conectado ao pino digital 13 deve acender, caso contrário, deve permanecer apagado.

Dica para leitura da porta serial:

```
// Verifica se há dados disponíveis para leitura
if (Serial.available() > 0) {
  // Lê o dado recebido
  dado = Serial.read();
  // Exibe o dado recebido no Monitor Serial
  Serial.print("Dado recebido: ");
  Serial.println(dado);
}
```

Dica de pausar a execução do programa até ter dados na porta serial:

```
while (Serial.available() == 0) {
  // Aguarda até que haja dados disponíveis
}
```

A screenshot of a code editor window titled 'Texto'. The code is written in C++ and is an Arduino sketch. It includes a `setup()` function that initializes `Serial` at 9600 baud. The `loop()` function checks if data is available on the serial port, reads the string, converts it to an integer using `atoi()`, and prints the result. The code is as follows:

```
1 int dadoInteiro = 0;
2 String dadoString = "";
3
4 void setup()
5 {
6     Serial.begin(9600);
7 }
8
9 void loop() {
10     // Verifica se há dados disponíveis para leitura na porta serial
11     if (Serial.available() > 0) {
12         // Lê a string recebida
13         dadoString = Serial.readStringUntil('\n');
14
15         // Converte a string para um número inteiro
16         dadoInteiro = atoi(dadoString.c_str());
17
18         // Exibe o número inteiro no Monitor Serial
19         Serial.print("Número inteiro recebido: ");
20         Serial.println(dadoInteiro);
21     }
22 }
```

atoi() é uma função da biblioteca padrão do C/C++ que converte uma string em um número inteiro (int). Aqui está uma explicação mais detalhada do que está acontecendo no código:

`dadoString.c_str()`: Este método retorna um ponteiro para o buffer interno que contém os caracteres da string `dadoString`. Este ponteiro é necessário como argumento para a função `atoi()`.

`atoi(dadoString.c_str())`: Esta função converte a string contida no ponteiro retornado por `dadoString.c_str()` em um número inteiro (int). Se a string não contiver nenhum dígito ou se não puder ser convertida em um número inteiro válido, `atoi()` retorna zero.