Oficina de introdução à robótica

Aula Online

Instrutores: Pedro Rocha e João Pedro Arruda
Organização: Profa. Regiane Kawasaki e Prof. Victor Santiago

/Aula Online

/Aula Online

Tópicos Principais:

1. Tipos de dados
2. Operadores
3. Funções
4. Const
5. Comentários

/Tipos de Dados

/Tipos de Dados

Inteiros (int):
Usados para representar números inteiros. Existem vários tipos de inteiros, como int, short, long e long long, que diferem em tamanho e faixa de valores.

Booleanos (bool):
Usados para representar valores verdadeiro (true) e falso (false). São comumente usados em expressões condicionais.

Ponto Flutuante (float e double):
Usados para representar números com casas decimais. O tipo float é de precisão simples, enquanto o double é de precisão dupla, oferecendo maior precisão.

/Tipos de Dados

Tipos de Estruturas (struct):
São usados para criar tipos de dados personalizados que podem conter várias variáveis de diferentes tipos.

Tipos de Classes (class):
Similar às estruturas, as classes também são usadas para criar tipos de dados personalizados, mas com a capacidade de encapsular dados e funções.

/Operadores

/Operadores

Operadores Lógicos:

&& E lógico **(AND)**

|| Ou lógico **(OR)**

! Não lógico **(NOT)**

Operadores de Incremento/Decremento:

++ Incremento

-- Decremento

```
int leituraA = digitalRead(sensorA);
int leituraB = digitalRead(sensorB);

if (leituraA == HIGH && leituraB == HIGH) {
  Serial.println("Ambos os sensores estão ativos.");
} else {
  Serial.println("Pelo menos um dos sensores está inativo.");
}
```

```
int estadoA = digitalRead(botaoA);
int estadoB = digitalRead(botaoB);

if (estadoA == HIGH || estadoB == HIGH) {
  Serial.println("Pelo menos um dos botões está pressionado.");
} else {
  Serial.println("Nenhum botão está pressionado.");
}
```

/Funções

/Funções

Como Definir?
Para definir uma função, é necessário primeiro declarar a função com o tipo de dado que ela irá retornar, o nome da função, após isso, se for necessário, colocar os parâmetros da função entre parênteses —Caso de trate de uma função void, ela não deve ser acompanhada de algum parâmetro—, assim então, pode abrir o escopo da função entre chaves. Sempre a lógica que é usada do escopo da função deve seguir a declaração que foi feita.

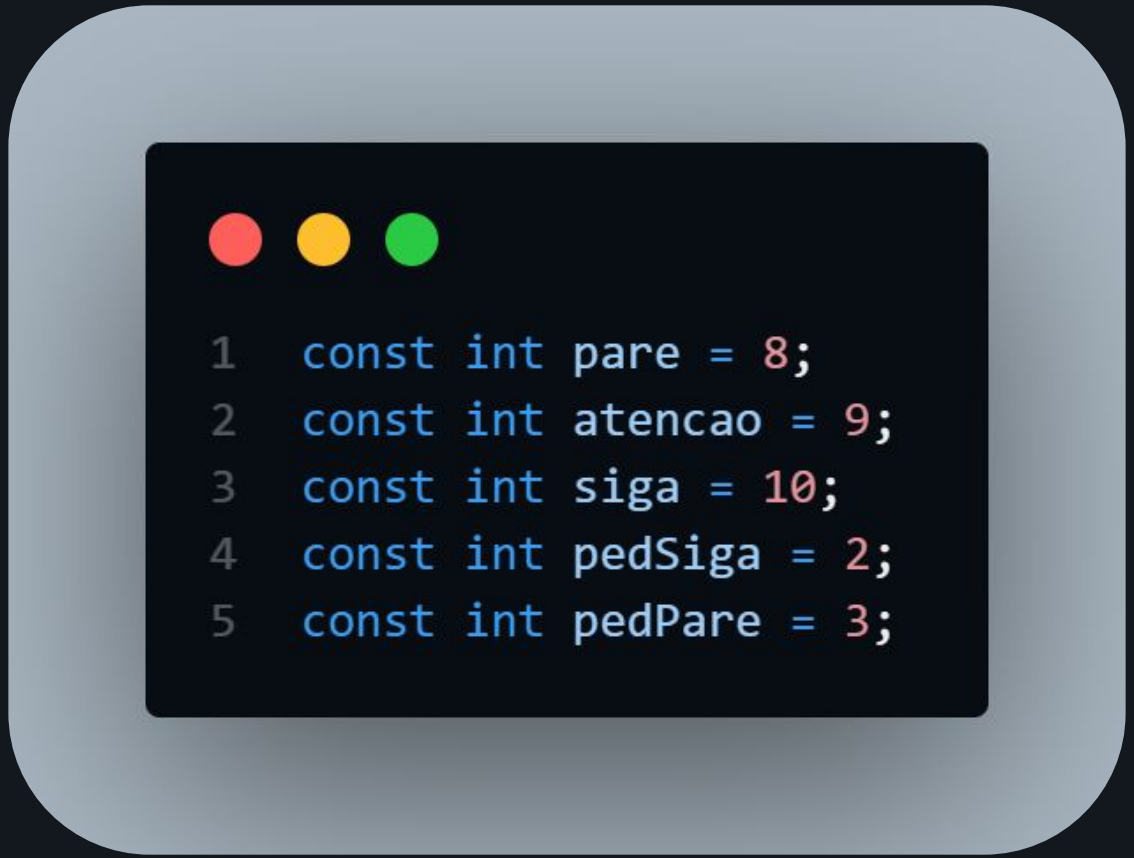
Chamada de Função:
Na chamada de função basta usá-la no código principal, escrevendo o nome da função acompanhado por parênteses com os parâmetros da função se for necessário.

/Const

/Const

Definição:
A palavra-chave const em C++ é usada para indicar que um objeto ou valor não deve ser modificado após sua inicialização. Ela desempenha um papel importante na prevenção de erros de programação. Ela pode ser usada em diversas partes do código, como na declaração de variáveis, parâmetros, classes, entre outros.

Como declarar:
Você pode declarar constantes usando a palavra-chave `const` antes do seu código. Isso é útil para tornar o código mais legível e para evitar que valores constantes sejam inadvertidamente alterados.


1 const int pare = 8;
2 const int atencao = 9;
3 const int siga = 10;
4 const int pedSiga = 2;
5 const int pedPare = 3;

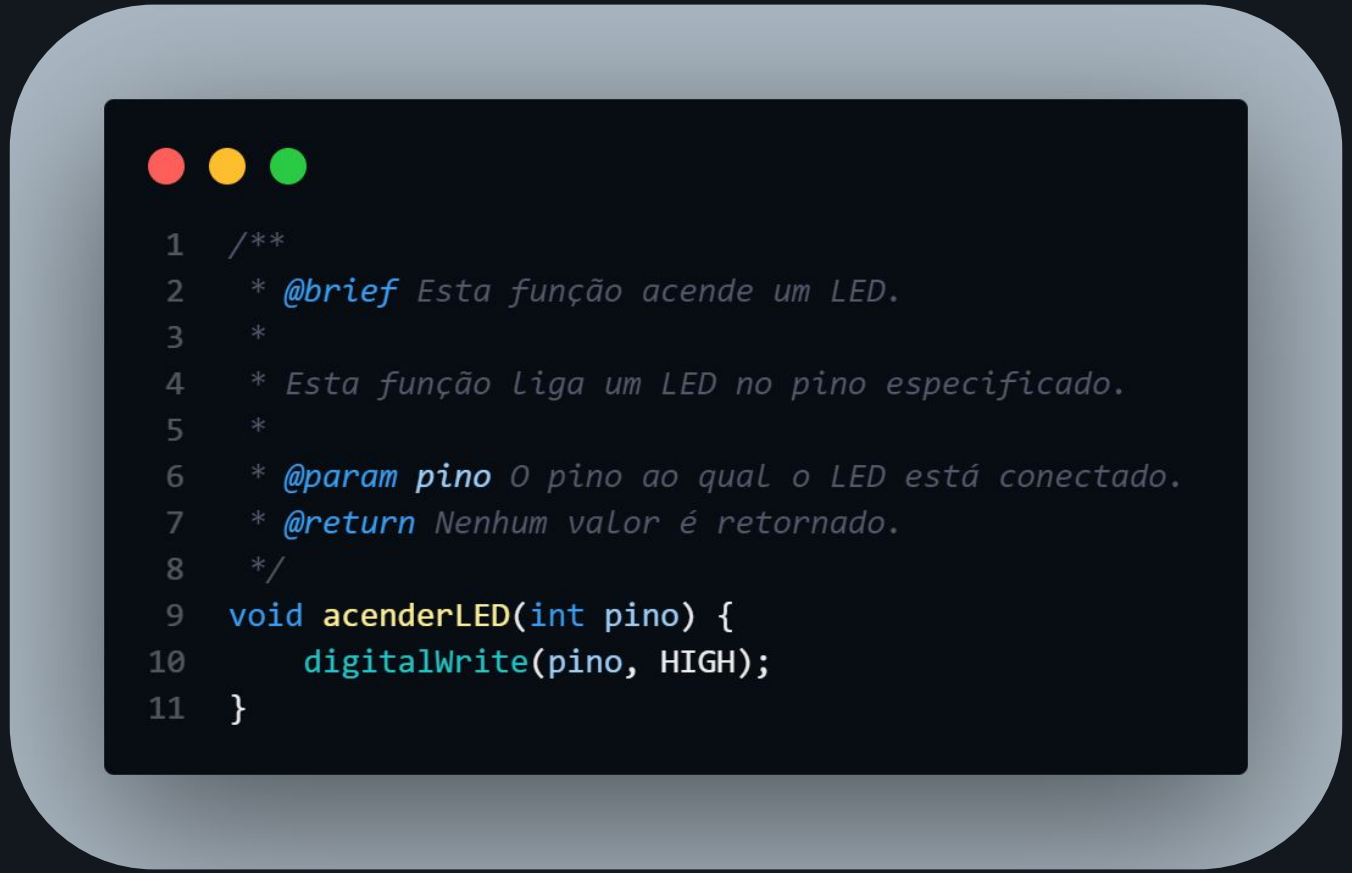
/Comentários

/Comentários

Linha Única:
Para adicionar um comentário de linha única em C++, você pode usar `'//'`. Qualquer texto após `'//'` em uma linha será tratado como um comentário e será ignorado pelo compilador.

Linhas Múltiplas:
Para criar comentários de várias linhas em C++, você pode usar `'/*'` para iniciar o comentário e `'*/'` para encerrá-lo. Qualquer texto entre esses delimitadores será considerado um comentário.

Comentários de Documentação:
Comentários de documentação começam com `'/**'` e podem conter informações detalhadas sobre classes, funções, parâmetros, retornos e outros elementos do código



```
/**
 * @brief Esta função acende um LED.
 *
 * Esta função liga um LED no pino especificado.
 *
 * @param pino O pino ao qual o LED está conectado.
 * @return Nenhum valor é retornado.
 */
void acenderLED(int pino) {
    digitalWrite(pino, HIGH);
}
```