

Templates em C++

Operador sizeof... em C++

Introdução

O operador `sizeof...` é utilizado para determinar o número de elementos em um pacote de parâmetros (parameter pack) em C++. Ele é útil para escrever templates que podem operar com um número variável de argumentos.

1. Definição e Sintaxe

- Definição: O operador `sizeof...` retorna o número de argumentos em um pacote de parâmetros.

- Sintaxe:

```
sizeof...(parameter_pack)
```

2. Exemplo Básico de Uso do sizeof...

- Definição: Um exemplo básico que demonstra o uso do operador `sizeof...` para contar o número de argumentos em um pacote de parâmetros.

- Exemplo:

```
template <typename... Args>
void contarArgumentos(Args... args) {
    std::cout << "Número de argumentos: " << sizeof...(args) << std::endl;
}
```

Templates em C++

```
int main() {  
  
    contarArgumentos(1, 2, 3);          // Saída: Número de argumentos: 3  
  
    contarArgumentos("Hello", "World"); // Saída: Número de argumentos: 2  
  
    contarArgumentos(1.0, 2.0, 3.0, 4.0); // Saída: Número de argumentos: 4  
  
    return 0;  
  
}
```

3. Uso em Classes Template

- Definição: O operador sizeof... pode ser usado em classes template para contar o número de parâmetros de template.

- Exemplo:

```
template <typename... Args>  
  
class Contador {  
  
public:  
  
    static void contar() {  
  
        std::cout << "Número de parâmetros de template: " << sizeof...(Args) << std::endl;  
  
    }  
  
};
```

```
int main() {  
  
    Contador<int, double, std::string>::contar(); // Saída: Número de parâmetros de template: 3  
  
    Contador<>::contar();                        // Saída: Número de parâmetros de template: 0  
  
    return 0;  
  
}
```

Templates em C++

4. Uso com Expansão de Pacotes de Parâmetros

- Definição: O operador `sizeof...` pode ser combinado com a expansão de pacotes de parâmetros para realizar operações em cada argumento do pacote.

- Exemplo:

```
template <typename... Args>
```

```
void imprimirArgumentos(Args... args) {
```

```
    std::cout << "Número de argumentos: " << sizeof...(args) << std::endl;
```

```
    (std::cout << ... << args) << std::endl;
```

```
}
```

```
int main() {
```

```
    imprimirArgumentos(1, 2, 3);           // Saída: Número de argumentos: 3, 123
```

```
    imprimirArgumentos("Hello", " ", "World"); // Saída: Número de argumentos: 3, Hello World
```

```
    return 0;
```

```
}
```

5. Uso em Construtores de Classe

- Definição: O operador `sizeof...` pode ser usado em construtores de classes template para inicializar objetos com um número variável de argumentos.

- Exemplo:

```
template <typename... Args>
```

```
class Variadico {
```

Templates em C++

public:

```
Variadico(Args... args) {  
    std::cout << "Número de argumentos: " << sizeof...(args) << std::endl;  
    (std::cout << ... << args) << std::endl;  
}  
};
```

```
int main() {
```

```
    Variadico<int, double, std::string> obj(1, 3.14, "texto"); // Saída: Número de argumentos: 3, 1
```

```
3.14 texto
```

```
    return 0;
```

```
}
```

6. Uso em Funções Recursivas

- Definição: O operador sizeof... pode ser usado em funções recursivas que processam pacotes de parâmetros.

- Exemplo:

```
void processar() {  
    std::cout << "Nenhum argumento restante" << std::endl;  
}
```

```
template <typename T, typename... Args>
```

```
void processar(T primeiro, Args... restantes) {
```

```
    std::cout << "Argumento: " << primeiro << ", Número restante: " << sizeof...(restantes) <<
```

Templates em C++

```
std::endl;

    processar(restantes...);

}

int main() {

    processar(1, 2, 3); // Saída: Argumento: 1, Número restante: 2
Argumento: 2, Número restante: 1
Argumento: 3, Número restante: 0
Nenhum argumento restante

    return 0;

}
```

Dicas de Boas Práticas

- Reutilização de Código: Use o operador sizeof... para escrever funções e classes que possam operar com um número variável de argumentos.
- Clareza e Manutenção: Mantenha o uso do operador sizeof... claro e bem documentado para facilitar a leitura e a manutenção do código.
- Verificação de Tipos: Verifique se os tipos dos parâmetros são os esperados para evitar erros de compilação ou execução.

Esta seção abrange os conceitos sobre o operador sizeof... em C++. Para mais detalhes, consulte a documentação oficial: <https://en.cppreference.com/w/cpp/language/sizeof...>