

Tipos de Cast

Prof. Valdir Pedrinho de Tomin Junior, Eng.

Universidade Federal de Santa Catarina
Centro Tecnológico de Joinville
EMB5631 – Programação III

e-mail: valdir.pedrinho@ufsc.br

16 de outubro de 2018

Sintaxe

```
static_cast <tipo> (expressão);
```

Características

- É uma conversão resolvida em tempo de compilação;
- Conversão implícita entre os tipos (como **float** pra **int**);
- Chama um construtor do **tipo** através do resultado da **expressão**;
- Usa operador de conversão definido pelo usuário do resultado da **expressão** para o **tipo**;
- Converte ponteiros entre hierarquia de classes, desde que as classes não sejam virtuais. (**static_cast** não verifica a validade da conversão durante a execução);
- Converte ponteiros **void*** para qualquer outro tipo de ponteiro.

Características

- As conversões implícitas facilitam lidar com variáveis numéricas e locais, mas é recomendado sempre explicitar a conversão quando exportando a variável para alguma função;
- **static_cast** pode ser usado para evitar a conversão implícita indesejada.

Exemplo

Sobrecarga da função f.

Sintaxe

```
dynamic_cast<tipo>(expressão);
```

Características

- Esta conversão é especial para referências ou ponteiros de objetos polimórficos. É dita dinâmica pois verifica durante a execução do programa se a conversão é válida quando descendo na hierarquia das classes;
- Ao converter ponteiros pra cima na hierarquia (**upcasting**) (**expressão** é derivada de **tipo**), comporta-se como uma conversão implícita;
- Ao converter ponteiros pra baixo na hierarquia (**downcasting**), (**expressão** é base de **tipo**), verifica se **expressão** originalmente referia-se a um ponteiro para **tipo** e, se sim, retorna o ponteiro ajustado. Caso a verificação falhe, retorna **nullptr**;
 - **nullptr** é um ponteiro para nulo, possui valor 0, assim como **NULL**, mas só é válido para atribuições de ponteiros.

Características

- **dynamic_cast** é a forma de verificar em tempo de execução se o tipo de objeto polimórfico passado é de um determinado tipo;
- **dynamic_cast** faz uso da **run-time type identification (RTTI)**, que é um recurso da linguagem de programação C++ que expõe informações sobre o tipo de dados de um objeto em tempo de execução.

Exemplo

Ponteiro para classe base.

Sintaxe

```
const_cast <tipo> (expressão);
```

Características

- Esta conversão tem a única função de adicionar ou remover a propriedade **const**;
- Qualquer conversão pode gerar uma referência ou ponteiro para um objeto para que este seja tratado como constante;
- Apenas **const_cast** pode gerar uma nova referência ou ponteiro para um objeto constante para que este seja tratado como modificável;

Características

- **const_cast** pode ser usado para alterar membros de classe não **const** dentro de uma função de membro **const**;
- **const_cast** pode ser usado para passar dados **const** para uma função que não recebe **const**;

Exemplo

Classe **student** e função **fun**

Sintaxe

```
reinterpret_cast <tipo> (expressão);
```

Características

- Conversão de ponteiro ou referência para qualquer tipo de objeto para ponteiro ou referência para qualquer outro tipo de objeto;
- Conversão de ponteiro para número inteiro;
- Conversão de número inteiro para ponteiro.

Potenciais perigos

Comanda o compilador a reinterpretar o resultado da **expressão** como se fosse do **tipo**, em geral sem realizar nenhuma operação nem verificação sobre os valores sendo convertidos;

Exemplo

Classe A