

Aula 7
Prof. Daniel Cavalcanti Jeronymo

Fundamentos de Programação

ET43G

Orientada a Objetos

Polimorfismo. Polimorfismo estático e dinâmico. Métodos virtuais, interfaces.

Universidade Tecnológica Federal do Paraná (UTFPR)
Engenharia Eletrônica – 3º Período
2016.1

- Polimorfismo
- Polimorfismos em C++
 - Estático e Dinâmico
- Métodos virtuais
- Classes abstratas e interfaces

Polimorfismo

- Polimorfismo
 - Objetos ou operações com diferentes comportamentos dependendo do contexto
- Objetos **iguais** exibem **diferentes** comportamentos
- Pode ser utilizado para estabelecer uma **interface**



Polimorfismo

[Strachey 1967]

Christopher Strachey, *Fundamental concepts in programming languages*, International Summer School in Computer Programming, Copenhagen, 1967.

- Definiu o polimorfismo ad-hoc e o polimorfismo paramétrico
- **Ad-hoc:** “*para isto*” em latim, “*funções que funcionam ou parecem funcionar para diferentes tipos (que podem não exibir uma estrutura comum)*” (Strachey, 1967)
- **Paramétrico:** “*funções que funcionam uniformemente para uma série de tipos, que normalmente exibem um tipo de estrutura*” (Strachey, 1967)

Polimorfismo

Computing Surveys, Vol 17 n. 4, pp 471-522, December 1985

On Understanding Types, Data Abstraction, and Polymorphism

Luca Cardelli

AT&T Bell Laboratories, Murray Hill, NJ 07974
(current address: DEC SRC, 130 Lytton Ave, Palo Alto CA 94301)

Peter Wegner

Dept. of Computer Science, Brown University
Providence, RI 02912

- Estenderam a definição de Strachey para incluir:

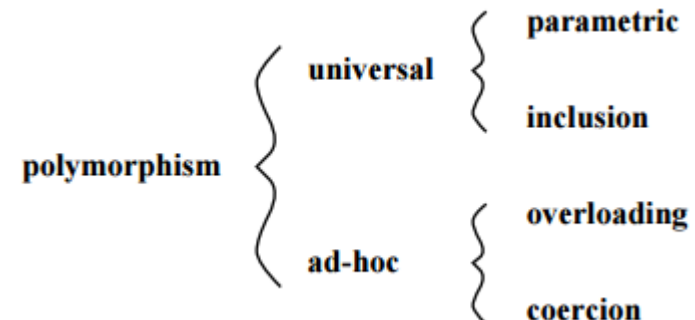


figura adaptada de Cardelli e Wegner (1985)

Os quatro tipos

- Quatro tipos principais de polimorfismo:
 - **Coerção (implícita ou explícita):** conversão (*cast*)
 - **Sobrecarga:** funções com assinaturas distintas e identificadores iguais
 - **Paramétrico:** utilização do mesmo código para diferentes tipos
 - **Inclusão / Subtipo:** em tempo de execução

Os quatro tipos

- Conversão:

```
double inteiro = 10; // inteiro (10) promovido para ponto flutuante  
int decimal = 9.99; // double (9.99) é demovido para inteiro
```

- Sobrecarga:

```
void imprime(int param);  
void imprime(char *param);
```

Todos estes são em tempo de compilação!

- Paramétrico: templates

próxima aula!

Os quatro tipos

- Inclusão / Subtipo
 - O comportamento diferente é determinado pelo subtipo do objeto
 - **Ocorre em tempo de execução!**
 - O objeto pode ser apresentado em termos de uma superclasse e ainda assim tem seu comportamento definido pela subclasse

Subtipo

- Inclusão / Subtipo

```
#include <iostream>
using namespace std;

void faz_acao(Animal &animal)
{
    animal.fala();
}

int main()
{
    Cachorro cachorro;
    Gato gato;

    faz_acao(cachorro);
    faz_acao(gato);

    return 0;
}
```

Subtipo

- Inclusão / Subtipo

```
class Animal
{
public:
    virtual void fala() { cout << " - vazio - " << endl; }
};
```

```
class Cachorro : public Animal
{
public:
    void fala() { cout << "au au!" << endl; }
};
```

```
class Gato : public Animal
{
public:
    void fala() { cout << "miau!" << endl; }
};
```

Subtipo

- Inclusão / Subtipo
 - **Problema:** a classe Animal pode instanciar um objeto, mesmo que isto não faça sentido
 - **Solução:** uso de uma classe base abstrata

```
class Animal
{
public:
    virtual void fala() = 0;
};
```

Subtipo

- Inclusão / Subtipo
 - **Como ocorre:** na linguagem C++ a solução típica é o uso de uma tabela que inclui os endereços das funções virtuais, conhecida como **tabela virtual (*v-table*)**
 - **Virtuais nulas:** a tabela apresenta um endereço nulo para objetos da classe base, o que impede (*deveria impedir*) a criação do objeto.

Subtipo

- Inclusão / Subtipo
 - **Virtuais nulas:** úteis para criar o que são conhecidas como **interfaces**, classes que modelam relações de **pode-fazer**



Exercício

- Modifique o exercício dos animais para que o mesmo apresente as quatro situações:
 - Método base não-virtual, método derivado não-virtual
 - Método base virtual, método derivado não-virtual
 - Método base não-virtual, método derivado virtual
 - Método base virtual, método derivado virtual
- Observe e anote o comportamento
- Anote também o tamanho do objeto resultante