



UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

BCC6002 - Aspectos de Linguagens de Programação

Prof. Dr. Rodrigo Hübner

Aula 05: reflexão, sobrecarga de métodos e operadores, metaclasses e closures

Reflexão

- Característica que permite no programa **observar** e também **modificar** a estrutura e o comportamento interno
 - Confunde-se com “**introspecção**”, no qual os dados são “apenas observados”
- Utilizados no desenvolvimento de **frameworks** e **testes** de software
- Interessante para **varrer** objetos e **serializá-los**
- Ver `reflection.cpp` e `reflection.cs`

Sobrecarga de métodos e operadores

- Possibilita **especificar o comportamento** de operadores nos tipos definidos pelo programador.

- Ajuda na legibilidade:

```
A * B + C * D
```

```
MatrixAdd(MatrixMul(A, B), MatrixMul(C,D))
```

- Algumas linguagens permitem **sobrecarregar os operadores padrões**: `C++`, `C#`, `Ruby`, `Python`, etc
- Ver `sobrecarga.cpp` e `sobrecarga_vetor.cpp`

Metaclasses

- É uma classe cuja a **instância é uma classe** (não chamada) e não promove os acessos ao objeto.
 - **Classes** definem o **comportamento de objetos**
 - **Metaclasses** definem o **comportamento de classes**
- Linguagens que suportam metaclasses:
 - `Common Lisp`, `Groovy`, `Object Pascal`, `Python`, `Perl`, `Ruby`, `Smalltalk`, etc
- Ver `metaclasses.py` e `metaclasses.dart`

Closures

- É uma ferramenta utilizada para “lembrar” e **acessar o escopo léxico de elementos** (funções, métodos, classes, ...) em qualquer parte de um programa
- Exemplo em `JavaScript`:

```
const add = (function () {  
  let counter = 0;  
  return function() { counter += 1; return counter; }  
})();  
add(); add();
```

- Ver `closure.js`

Próxima aula

- **Desenvolvimento do trabalho e revisão de prova**