

Aula 2 - Programação Funcional em Coq

Rafael Castro - rafaelcgs10.github.io/coq

02/05/2018

O que é Programação Funcional?

- O paradigma de programação funcional é baseado no simples conceito: se funções não têm efeitos colaterais, então tudo o que precisamos saber sobre funções são os seus mapeamentos.
- (se ignorarmos propriedades não funcionais como a eficiência!).
- O termo funcional enfatiza que funções são *first class values*, ou seja, elas podem ser passadas como argumento para funções e retornadas por funções.

Baby's First Functional Program

- Um programa funcional é apenas um mapeamento!
- Um programa (não recursivo) é definido com *Definition*, seguido pelo identificador, a sequência de argumentos, o tipo do seu retorno e a sua definição.
- Casamento de padrão desconstrói um identificador em seus possíveis termos. A seta \Rightarrow informa o respectivo resultado.

```
Definition next_weekday (d:day) : day :=
  match d with
  | monday => tuesday
  | tuesday => wednesday
  | wednesday => thursday
  | thursday => friday
  | friday => monday
  | saturday => monday
  | sunday => monday
end.
```

Executando computações em Coq

```
Compute (next_weekday friday).
```

```
Compute (next_weekday (next_weekday saturday)).
```

Baby's First Proof about some Function

- *Example* é o mesmo que *Theorem*, que também é o mesmo que *Lemma*.

Example test_next_weekday:

(next_weekday (next_weekday saturday)) = tuesday.

Proof.

simpl. reflexivity.

Qed.

Tipo booleano

- O tipo padrão booleano é definido em Coq por:

```
Inductive bool : Type :=
| true  : bool
| false : bool.
```

- Como isso, de alguma, maneira representa os booleanos? De forma alguma!
- Booleanos tem sua semântica dada pelas suas operações (tabela verdade).

Funções sobre booleanos

