

Interpretadores

Prof. Vander Alves

Linguagem de Expressões 1

- Inclui apenas constantes (valores) e operações sobre valores
- Valores e operações sobre inteiros são admitidos
- Um programa é uma expressão

Exemplos

- $22 * (2+3)$
- $(123 + 47 - 6) * 222 / 4$
- ...

Gramática (Sintaxe Concreta)

EAdd. $\text{Exp} ::= \text{Exp} \text{ "+" } \text{Exp1} ;$

ESub. $\text{Exp} ::= \text{Exp} \text{ "-" } \text{Exp1} ;$

EMul. $\text{Exp1} ::= \text{Exp1} \text{ "*" } \text{Exp2} ;$

EDiv. $\text{Exp1} ::= \text{Exp1} \text{ "/" } \text{Exp2} ;$

EInt. $\text{Exp2} ::= \text{Integer} ;$

coercions $\text{Exp } 2 ;$

Tipo Algébrico (Sintaxe Abstrata)

```
data Exp
  = EAdd Exp Exp
  | ESub Exp Exp
  | EMul Exp Exp
  | EDiv Exp Exp
  | EInt Integer
deriving (Eq, Ord, Show, Read)
```

Interpretador

```
eval :: Exp -> Integer
```

```
eval x = case x of
```

```
    EAdd exp0 exp  -> eval exp0 + eval exp
```

```
    ESub exp0 exp  -> eval exp0 - eval exp
```

```
    EMul exp0 exp  -> eval exp0 * eval exp
```

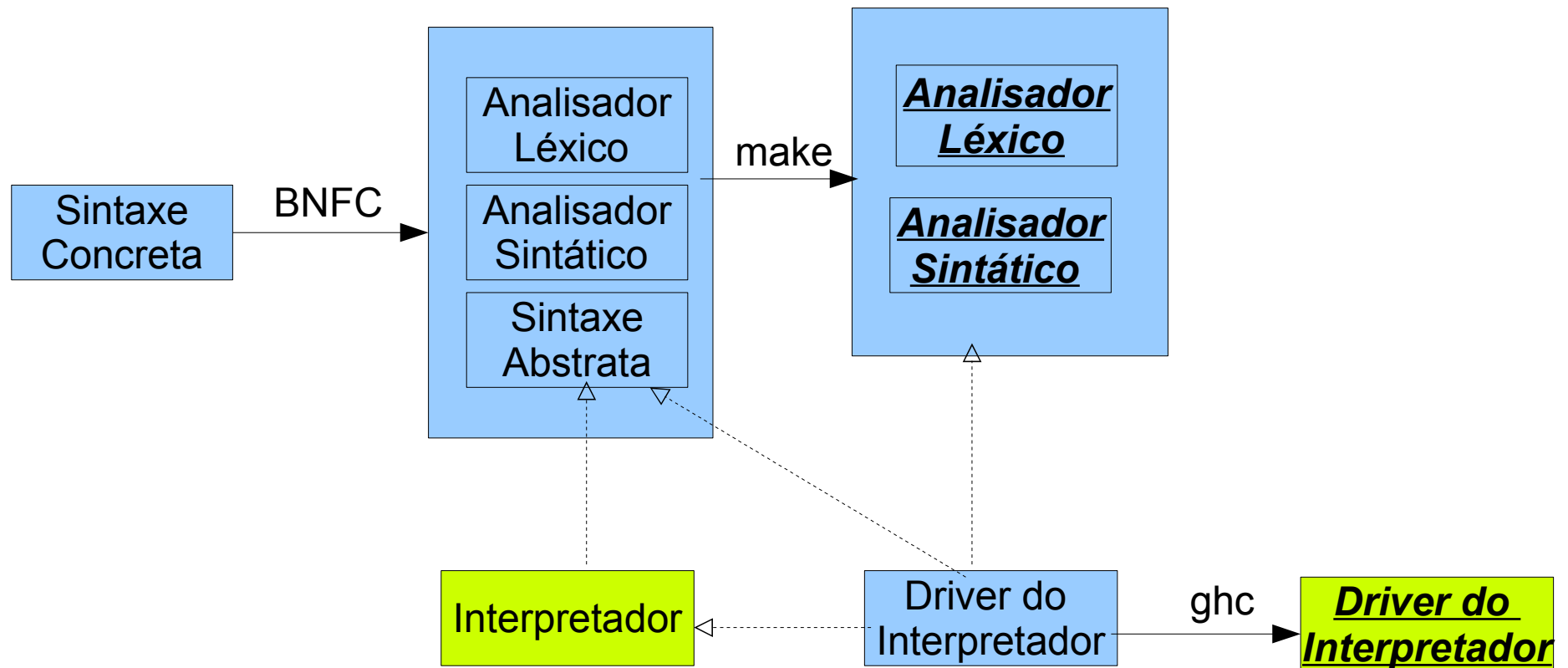
```
    EDiv exp0 exp  -> eval exp0 `div` eval exp
```

```
    EInt n    -> n
```

Demo

- Instalar plataforma Haskell
- Instalar BNF Converter (BNFC)
 - <http://bnfc.digitalgrammars.com/>
 - Ler tutorial
 - <http://bnfc.digitalgrammars.com/tutorial.html>
- (Instalar make)
- Editar gramática -> BNFC -> make -> editar interpretador -> compilar interpretador (*driver*)

Processo



Exercício

- No Moodle, descompacte o arquivo **LE1.rar** e estude a execução do processo apresentado no slide anterior com base no arquivo **README.txt**