

Interpretadores

Linguagem Imperativa 2

Prof. Vander Alves

Linguagem Imperativa 2

- LI2 = LI 1 estendida com definição de funções e expressões booleanas e strings
- Um programa é uma lista de definição de funções
- A execução do programa começa pela função **main**

Exemplos

```
main () {  
    z = 1;  
    w = "hello";  
    v = "world";  
    a = w ++ v;  
    return a;  
}
```

Exemplos

```
main () {  
    a = true;  
    b = false;  
    e = true;  
    d = !a || b && e;  
    f = a && b || ! e;  
}
```

Exemplos

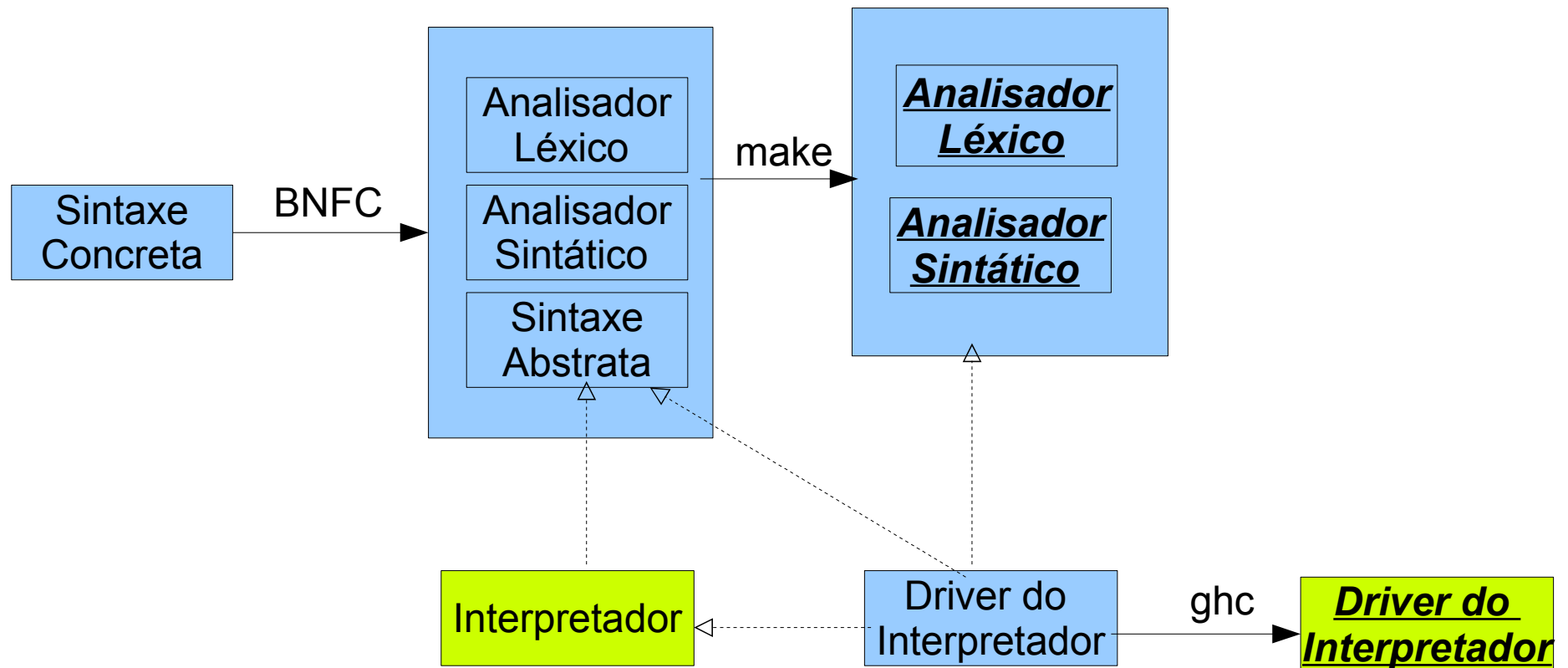
```
main () {  
    f = fat (5) ;  
    return f ;  
}
```

```
fat (n) {  
  
    if (n)  
        then return n * fat (n-1) ;  
        else return 1 ;  
  
}
```

Projeto e Implementação

- Sintaxe concreta
- Sintaxe abstrata
- Interpretador
- Vide código (arquivo **LI2.rar** no Moodle)

Processo



Exercícios

- Obtenha o valor do tipo algébrico representando cada programa apresentado como exemplo da LI2 nos slides anteriores
 - Você pode checar o resultado usando o programa **TestLI**, que faz parte do arquivo **LI2.rar**
- Estenda a **LI2** (tipos algébricos e interpretador) para implementar variáveis globais