

# Compiladores: Projeto

Zhao Liang

2023-10-01

## 1 A linguagem XYZ

### 1.1 Descrição

- Uma linguagem imperativa com a sintaxe parecida com a linguagem C;
- Um programa XYZ é uma sequência de definições de funções sem recursividade e sem variáveis globais;
- A função principal, por onde é iniciada a execução das instruções, chama-se **main**, retorna um valor e não possui parâmetros;
- Só há dois tipos de dados na linguagem, o tipo **inteiro** de 64 bits (**i64**) e o tipo ponto flutuante de 64 bits (**f64**). Na avaliação Booleana o 0 é falso e qualquer número diferente de 0 corresponde ao Booleano verdadeiro.
- Espaço em branco ‘ ’, tabulação ‘\t’, retorno de “carro” ‘\r’ e nova linha ‘\n’ devem ser ignorados.

**Restrições.** A linguagem é bem restrita, não havendo vetores, ponteiros e módulos. Todo o programa é definido em um único arquivo.

**Comentários.** Todos os comentários devem começar com “//”.

### 1.2 Funções

#### Sintaxe

- Uma definição de função começa com a palavra reservada **fn** seguida pelo seu **nome**, uma lista de parâmetros separados por vírgula entre parênteses e um corpo.
- Todas as funções devem retornar um valor mesmo que este valor não seja usado diretamente. O retorno do valor é realizado usando o comando **return**.
- Os parâmetros de uma função devem ser separados por vírgula. Exemplo:  
`fn foo(a i64, b f64) { ... }`

## Proposição (*statement*)

Uma proposição na linguagem XYZ pode ser:

1. proposição vazia;
2. declaração de variáveis (*declarations*);
3. atribuição (*assignment*);
4. incremento e decremento;
5. retorno de função;
6. chamada de função (*function call*);
7. desvio de fluxo (**if** com e sem **else**), sem suporte a **else if**;
8. laço (*loop*, só há um, o **while**).

**Identificadores** (*identifiers*). Um identificador (nome) é uma letra, opcionalmente seguida por letras e *underscores* “\_”.

**Declarações.** Uma variável deve ser declarada antes de ser usada começando pela palavra reservada **var** e uma lista de atribuições separada por vírgula “,”. Nenhuma variável pode ser declarada sem possuir um valor inicial.

## Expressões

As seguintes expressões existem em XYZ:

- Variáveis (**if a {...}** ) e literais (**while 1 {...}**);
- Expressões binárias com os operadores:  
+ - \* / % > < >= <= == != && ||;
- Expressões unárias com os operadores: - !;
- Chamadas de função.

A Listagem 1 mostra o código para o cálculo do fatorial como um exemplo de aplicação da linguagem XYZ.

```
1 // fat.xyz
2 fn fatorial(n i64) {
3     var
4         i : i64 = 1,
5         r : i64 = 1;
6
7     if n < 0 {
8         return -1;
9     }
10
11     while i < n + 1 {
12         r = r * i;
13         i++;
14     }
15     return r;
16 }
17
18 fn main() {
19     var
20         i : i64 = 3,
21         f : i64 = 0;
22
23     f = fatorial(i);
24
25     return 0;
26 }
```

Listagem 1: Exemplo de código usando a linguagem XYZ.

## Projeto

A partir da definição da linguagem XYZ:

1. Escreva um analisador léxico usando o `lex` para a linguagem;
2. Escreva um analisador sintático usando o `yacc`;
3. A partir do analisador sintático, imprima a tabela de símbolos para o código de entrada com os símbolos apresentados de acordo com o contexto. Por exemplo, para a Listagem 1 a saída da execução do compilador seria parecida com:

```
fatorial.n [i64]
fatorial.i [i64]
fatorial.r [i64]
main.i [i64]
main.f [i64]
```

4. Descreva como checar se uma variável ao ser usada, já foi declarada. (Não é necessário especificar na gramática.)

O código da Listagem 1 pode ser usado como referência nos testes. Se o arquivo contendo as regras léxicas chama-se `xyz.l`, com as regras gramaticais `xyz.y` e o código da Listagem 1 `fat.xyz`, os seguintes comandos podem ser executados:

```
1 lex -o xyz.yy.c xyz.l
2 yacc -d -o xyz.tab.c xyz.y
3 cc -o xyz xyz.tab.c
4 ./xyz fat.xyz
```

Quaisquer regras gramaticais que possam surgir e que não estejam definidas neste documento, podem ser arbitradas pelo desenvolvedor da gramática.

Quaisquer dúvidas podem ser postadas como *issue* no repositório das práticas da disciplina, para que todos tenham acesso às informações.