

Construção de Compiladores

Guia da Linguagem MiniC

1. Introdução

Este guia descreve a linguagem de programação MiniC, que será usada como linguagem fonte para o projeto de um compilador da disciplina. MiniC é um subconjunto da linguagem C, ou seja, todo programa MiniC é também um programa C, mas há muitas características da linguagem C completa que não aparecem em MiniC. Por exemplo, existem poucos tipos básicos, e não é possível escrever o programa usando mais de um arquivo fonte. Não existem protótipos: cada função usada pela função main deve ser definida antes desta.

Aqui está descrita a estrutura léxica da linguagem. Ao final apresentamos um programa de exemplo simples em MiniC.

2. Estrutura Léxica

A estrutura léxica da linguagem é simples. As classes léxicas (tipos dos tokens) são:

- Palavras-chave
- Identificadores
- Literais Inteiro (int)
- Literais de Ponto flutuante (double, float)
- Literais string
- Literais char
- Operadores
- Símbolos de Pontuação
- Comentários

Palavras-chave: Estas são palavras reservadas que não podem ser usadas como identificadores. As palavras-chave do MiniC são:

- **char**
- **else**
- **if**
- **int**
- **float**
- **double**
- **main**
- **printf**
- **printint**
- **printstr**
- **return**
- **static**
- **void**
- **while**
- **for**
- **do**
- **include**

Identificadores: os identificadores seguem a mesma regra da linguagem C: podem começar com uma letra ou sublinhado ('_'); os demais caracteres podem ser letras, sublinhado ou dígitos.

Literais Inteiros: são literais inteiros, ou seja, cadeias de dígitos numéricos. Por exemplo, 1; 24; 4567, entre outros.

Literais de Ponto flutuante: são literais reais, como por exemplo, 2.55, 45.678, 0.75, entre outros.

Literais caracteres: seguem a regra da linguagem C, sendo compostos por um caractere envolto em aspas simples. O caractere pode ser uma das sequências de escape '\r', '\n' ou '\t'. Para representar uma barra invertida como caractere, deve-se usar '\\'.

Literais String: mesma regra da linguagem C: começam e terminam com um caractere de aspas duplas, e podem conter as sequências de escape \r, \n e \t. Para incluir uma

barra invertida (\) na string, deve-se usar \\. Uma string representa um array de caracteres. Por exemplo: “\nHello World\n”.

Operadores: os operadores binários em MiniC são as quatro operações aritméticas (+, -, * e /), os operadores de comparação (==, !=, <, >, <= e >=) e os lógicos (&& e ||); há também o operador de atribuição (=). O único operador unário é o de negação lógica (!).

Símbolos de Pontuação: os símbolos de pontuação na linguagem MiniC são os tipos básicos encontrados na linguagem C, sendo eles : ; , () { } . # &

Comentários: seguem as mesmas regras da linguagem C: comentários até o final da linha começando com // ou comentários multilinha começando com /* e terminando com */. Comentários multilinha não podem ser aninhados.

3. Exemplos

A Figura a seguir apresenta um exemplo simples de programa que calcula o fatorial do número escrito na linguagem MiniC.

```
#include <stdio.h>
#include <stdlib.h>

// Programa Teste Projeto Compiladores

int main()
{
    printint(fatorial(5));

    return 1;
}

int fatorial(int n)
{
    int res;
    char n = 'i';
    char i = '\n';

    if (n < 1)
        res = 1;
    else
        res = n * fatorial(n - 1);

    return res;
}
```

Os tokens devem ser gerados seguindo a estrutura: <tipo, lexema, linha, coluna>.

4. Informações Importantes

Data Entrega: 16/09/2018 (via Unipê virtual)

Equipe: No máximo 3 pessoas.

Ferramenta: JFlex (Java).

Observação. Ao final do trabalho, cada aluno deverá fazer o upload do **projeto completo**, incluindo:

- a) O arquivo de especificação (.flex).
- b) A classe do analisador léxico gerado pelo JFlex.
- c) A classe que gera o analisador léxico a partir do arquivo de especificação.
- d) A classe principal que utiliza o analisador léxico gerado passando um arquivo de teste como entrada.

Plágio: Trechos de código não deverão ser compartilhados entre diferentes duplas, pois trabalhos copiados receberão **nota 0 (zero)**.

Nota: 3,0.

5. Notas Bibliográficas

A linguagem MiniC foi inicialmente baseada na linguagem MiniJava, apresentada no livro de Appel sobre compiladores em Java [1].

Referências

[1] Appel, Andrew W. e Palsberg, Jens, Modern Compiler Implementation in Java, 2nd ed., (Cambridge: Cambridge University Press, 2002).