

Desenvolvimento do Compilador

Compiladores

Prof. Charles Ferreira

1 Instruções

Este trabalho deverá ser desenvolvido em grupos de, **no máximo**, 5 alunos;

Forma de entrega: eletrônico;

A **versão eletrônica** deve conter: o código fonte, a aplicação executável e códigos da linguagem de cada grupo com a linguagem Java;

Data de entrega: **10/11/2020**;

Durante a aula do dia **10/11/2020** cada grupo deverá apresentar a Linguagem de Programação implementada.

2 Enunciado

Cada grupo deve definir a sua própria gramática e todos os tokens necessários. Os requisitos mínimo são:

- Deve ter 3 tipos de variáveis
- Deve ter a estrutura de controle if ... else;
- Deve ter duas estruturas de repetição (while, do ... while, for);
- A parte de expressões envolvendo os operadores matemáticos deve ser realizada de maneira correta, respeitando a precedência. Não é necessário gerar a resposta da expressão, basta cuidar da precedência entre os operadores matemáticos através da gramática;
- As atribuições também devem ser realizadas;
 - É necessário verificar se é possível realizar as operações, devido aos tipos das variáveis e ao seu escopo.

- Os comandos de leitura do teclado (Scanner) e de impressão na tela (println) devem ser disponibilizados.
- O compilador tem que aceitar números decimais.
- A cada utilização de uma variável, é necessário verificar se ela já foi declarada.

O compilador deve fazer a conversão de um programa desenvolvido na Linguagem definida pelo grupo para a Linguagem Java (fica a critério do professor definir para qual linguagem).

A verificação da corretude do programa será realizada compilando o arquivo gerado pelo compilador desenvolvido.

Seu compilador deverá receber como entrada um arquivo contendo um programa escrito na Linguagem definida pelo grupo e gerar uma forma equivalente em Java, que deverá ser compilada no compilador javac, executada na JVM e não deverá conter erros.

OBS: a gramática não pode conter recursividade à esquerda e produções vazias (que porventura venham a surgir). Caso seja necessário, efetue a sua fatoração à esquerda.

3 Exemplo de um Compilador

A descrição a seguir ilustra um exemplo de um Compilador que faz a conversão de um programa desenvolvido em uma linguagem fictícia para uma forma equivalente na linguagem C.

Os termos em **negrito** significam palavras reservadas. Preste atenção aos sinais de pontuação.

Prog → **programa** Declara Bloco **fimprog**.

Declara → (**inteiro** | **decimal**) Id (, Id)*.

Bloco → (Cmd)+

Cmd → CmdLeitura | CmdEscrita | CmdExpr | CmdIf

CmdLeitura → **leia** '(' Id ')'

CmdEscrita → **escreva** '(' Texto | Id ')'

CmdIf → **if** '(' Expr Op_rel Expr ')' '{' Cmd+ '}' (**else** '{' Cmd+ '}')?

CmdExpr → Id ':' '=' Expr.

Op_rel → '<' | '>' | '<' '=' | '>' '=' | '!' '=' | '=' '='

Expr → Expr '+' Termo | Expr '-' Termo | Termo

Termo \rightarrow Termo '*' Fator | Termo '/' Fator | Fator

Fator \rightarrow Num | Id | '(' Expr ')'

Texto \rightarrow ' "' (0-9 | a-z | A-Z | ' ')+ ' "'

Num \rightarrow ([0-9])+

Id \rightarrow ([a-z] | [A-Z])([a-z] | [A-Z] | [0-9])*

OBS: espaços em branco, tabs e enter podem aparecer e devem ser eliminados.

<pre>programa inteiro a, b, c. decimal d. escreva("Programa Teste"). escreva("Digite A"). leia(a). escreva("Digite B"). leia(b). if(a<b) { c:= a + b. }else{ c:= a - b. } escreva("C e igual a "). escreva(c). d := c / (a + b). escreva("D e igual a "). escreva(d). fimprog.</pre>	<pre>#include <stdio.h> void main(void) { int a,b,c; double d; printf("Programa Teste"); printf("Digite A"); scanf("%d",&a); printf("Digite B"); scanf("%d",&b); if(a<b) { c=a+b; }else { c = a-b; } printf("C e igual a "); printf("%d",c); d = c / (a + b); printf("D e igual a "); printf("%lf", d); }</pre>
---	---