**DOCUMENTAÇÃO**

O projeto é dividido em:

**1º Parte** - definição de tokens

É definido da forma seguinte:

#define \_NOME\_DO\_TOKEN valor\_de\_retorno

significado das siglas:

\_OPMULT - operador de multiplicacao

\_OPADI - operador de adicao

\_OPSUB - operador de subtracao

\_PNTVRGL -

\_DSPNT -

\_ABREP -

\_FECHAP -

\_PNT -

\_VRGL -

Cada token possui um valor inteiro que será usado de retorno para cada reconhecimento do automato.

**2º Parte** - O automato finito - reconhecedor de palavras

Aqui temos um dispositivo de reconhecimento de palavras reservadas que receberá uma string "str" e uma posicao “i" de início. O automato incrementará o valor do “i" até identificar um espaço em branco(limitador) para retornar o valor inteiro do token(define), caso a palavra não for reconhecida por uma sequencia incorreta o automato retornará ZERO e finalizará o reconhecimento.

Se a string de entrada "str" na posicao “i” possuir um espaço em branco o automato retornará -1. Caso ela possua quebra de linha o automato retornará 1. Isto será usado posteriormente para limitar fim da linha na leitura do arquivo.

**3º Parte** - main

**Leitura** do arquivo “program.txt”.

A leitura é feita linha por linha chamando o método leitor() reconhecendo as palavras contidas. Quando uma palavra não é reconhecida a execução é finalizada.

**Saída do programa** é feita da seguinte forma:

A cada palavra reconhecida o formato seguinte é exibido:

r1: <alguma\_coisa>

r2: <outra\_coisa>

Token:<valor\_do\_token, mais\_outra\_coisa>

Caso a palavra não foi reconhecida:

r3: <outra\_coisa>

A **escrita** do arquivo “saida.txt” contém somente os tokens reconhecidos, um em cada linha.