%{

#include <stdlib.h>

#include "e1.tab.h"

#define DEBUG 1

%}

#define DEBUG – Se for informado 1 mostra os caracteres lidos do arquivo de input (exemplo.e1). Então se eu puser #define DEBUG 0 não mostrará os caracteres lidos do arquivo de input.

Os dois includes permitem o uso da função atoi (stdlib.h) e o include “e1.tab.h” traz os tokens declarados no parser/analisador sintático (e1.y) para que o lexer/analisador léxico (e1.l) possa usar.

%%

"+" |

"-" |

";" |

"\*" |

"/" |

"(" |

")" |

"=" {

#if DEBUG

printf("token com um caracter de valor %c na linha %d\n", yytext[0], yylineno);

#endif

return yytext[0];}

Não foram declarados ‘token class’ para os símbolos acima, preferindo-se retornar o próprio símbolo (return yytext[0];).

Se optasse por fazê-lo, seria algo assim (sendo que PLUS teria sido previamente declarado com %token PLUS no analisador sintático):

“+” {return PLUS;}