**Nome do software:** Analisador Léxico

**Integrantes do grupo:** Lucas Garavaglia

**Descrição da linguagem:** C é uma linguagem de programação compilada de propósito geral, estruturada, imperativa, procedural, padronizada pela Organização Internacional para Padronização (ISO), criada em 1972 por Dennis Ritchie na empresa AT&T Bell Labs para desenvolvimento do sistema operacional Unix (originalmente escrito em Assembly).

**Descrição do Autômato:** Foi criado um autómato universal que lê e executa uma sequência de estados previamente declarados. Foram criadas 12 classes de estados referente a cada classe de token.

**Funcionamento do software:** O software funciona com base em autômatos, assim que iniciar o programa com o comando npm start dentro do diretório principal irá aparecer uma mensagem com orientações, siga a orientação colocando o caminho com o nome do arquivo exemplo: /home/compiladores/trabalho/src/fonte1.txt, após inserido o caminho ele era ler e fazer a análise léxica, se houver algum erro ele irá indicar, caso contrário ele finaliza deixando uma mensagem de “Concluído !”

**Erros:** Após uma breve pesquisa foi definido os seguintes tipos de erros: palavra/token não pertence a nenhuma classe de tokens válidas, identificador mal formado e número mal formado.

**Build:** Para o funcionamento do código não será preciso compilar, somente usar o comando “npm start” dentro do diretório principal do projeto, para o desenvolvimento do trabalho foi utilizado o editor de texto VScode e a linguagem javascript com a framework nodejs.

**Classe de tokens:**

Token para marcação: [{,}]

Token para função: [(,)]

Token para Loop: [while]

Token para condição: [if]

Token para final de função: [return]

Token para tipo de dado: [int]

Token para atribuição: [=]

Token para Expressões: [>=,<=,==,!=]

Token para operadores: [=,-,\*,/,%]

Token para final de linha: [;]

Token para Caracteres: [A-Z,a-z]

Token para números: [0-9]