

Alunos: Fernanda Das Neves Merqueades Santos, Giovanna Rodrigues Mendes, Matheus Felipe Alves Durães e Roney Felipe De Oliveira Miranda.

RGAs: 2021.1904.035-1 ; 2021.1904.032-7 ; 2021.1904.008-4 ; 2021.1904.030-0.

Grupo: Loucos, Fracos e Atormentados (LFA).

Objetivo do grupo: Implementação dos comandos switch - case + comando break.

Funções criadas: F_CASE, CASE_, C_, EXP_CASE, QUEBRAR

Relatório

Passo a passo: Primeiramente, escolhemos o lex2 como método léxico para nossa implementação. Com isso, inserimos o comando “case” nas keywords e terminalName dentro do lex2.c e lex.h na sua posição referente seguindo a ordem alfabética. Tendo isso feito, começamos a construir as produções dentro do parser.c, onde ao todo foram elaboradas 5 produções, como pode ser visto abaixo:

```
/* F_CASE -> case (C_): CASE_ */
/*      | case C_: CASE_ */
/*      | default: C */
/*      | Epsilon */
/* Producao incubida por analisar o case,
tanto o termo quanto a condicao, ou default*/
```

```
/* CASE_ -> F_CASE */
/*      | QUEBRAR F_CASE */
/*      | C QUEBRAR F_CASE */
/* Producao responsavel por ler o corpo do case,
ela não analisa o termo CASE responsavel pela chamada de
F_case() e nem a condicao de tal. */
```

```
/* C_ -> id EXP_CASE */
/*      | num EXP_CASE */
/* Producao responsavel por verificar se o que foi lido
e uma variavel ou um valor real.*/
```

```
/* QUEBRAR -> break; */
/*      | Epsilon */
/* Producao responsavel por verificar todos os
breaks compilados*/
```

```
/* EXP_CASE -> +C_ */
/*      | *C_ */
/*      | -C_ */
/*      | /C_ */
/*      | Epsilon */
/* Producao responsavel por verificar os operando */
```

Paralelamente, adicionamos os cabeçalhos das produções criadas no parser.h. Todo o código, assim como a documentação por nós gerada durante o desenvolvimento do trabalho, está disponível no GitHub¹.

¹ Link para repositório no GitHub: https://github.com/RoneyFelipe/Trabalho_LFA