Alunos: Rodolfo Claudio Cruz e Mateus Campello

**Introdução**

O algoritmo escolhido é o verificador de presença na sequência Fibonacci. Ao ser executado, o programa pede um ou mais valores de entrada e retorna uma lista com uma descrição de verdadeiro ou falso se o valor aparece na sequência ou não. Somente valores naturais são aceitos, e para isso, foram criadas regras que serão descritas abaixo.

**Código**

Abaixo temos as regras do programa lex. A primeira impede que o programa aceite valores negativos, a segunda impede que aceite valores reais, a terceira aceita um valor único como entrada, a quarta regra aceita múltiplos valores na mesma entrada, com a vírgula sendo usada como separador, e a quinta e última regra para encerrar o programa.

/\* Rules \*/

**%%**

-{digits}+ {

    printf("Negative number is invalid!\n");

}

{digits}+.{digits}+ {

    printf("Only positive integers are valid!\n");

}

^{digits}+$ {

    number = atoi(yytext);

    is\_fib = number\_is\_fib(number);

    printf("\nDoes the number below occur in the Fibonacci sequence?\n\n");

    print\_result(is\_fib);

}

(?:{digits}+,)+{digits}+$ {

    token = strtok(yytext, delim);

    printf("\nDo the numbers below occur in the Fibonacci sequence?\n\n");

    while (token != NULL) {

        number = atoi(token);

        is\_fib = number\_is\_fib(number);

        print\_result(is\_fib);

        token = strtok(NULL, delim);

    }

}

quit|q|exit {

    printf("-------------------------------------------------- EXITING... --------------------------------------------------\n");

    exit(1);

}

. ;

**%%**

/\* End Rules \*/

E aqui temos a função de verificação de aparição na sequência Fibonacci. É uma função que compara o valor recebido com os dois valores iniciais da sequência, e então, iterativamente, compara com todos os valores seguintes da sequência, retornando 1 e encerrando o loop caso o valor seja encontrado na sequência, ou 0 caso não seja.

int number\_is\_fib(int n) {

    seq = 0;

    a = 0;

    b = 1;

    if (n == a || n == b)

        return 1;

    seq = a + b;

    while(seq <= n && is\_fib == 0) {

        if (seq == n)

            return 1;

        a = b;

        b = seq;

        seq = a + b;

    }

    return 0;

}

E por fim, um exemplo de uso do programa:

-------------------------------------------------------------------------

------------------------------ WELCOME!! --------------------------------

-------------------------------------------------------------------------

--> INSTRUCTIONS: -------------------------------------------------------

-> To evaluate a positive integer, type it and then press ENTER

-> To evaluate more than one positive integer, type each one separated by a comma and then press ENTER

-> To exit the application, you can type exit, q, or quit

-------------------------------- ENJOY!! --------------------------------

48,876,2,4185,5,658

Do the numbers below occur in the Fibonacci sequence?

48 - No

876 - No

2 - Yes

4185 - No

5 - Yes

658 – No

q

-------------------------------- EXITING... -----------------------------

Repositório com os códigos-fonte: [GitHub](https://github.com/Vasllo/TrabalhoDeCompiladores)