

1 Projeto – Parte1

1.1 Descrição Geral

Desenvolva um programa (de preferência em **Java**) que realize as operações de uma **fórmula lógica**, no formato de **tabela-verdade**.

1.2 Requisitos

Entrada: String contendo uma fórmula lógica – a String deve ser passada via **teclado** pelo usuário

Operações lógicas permitidas: \wedge , \vee , \neg (apenas operações canônicas)

Número de variáveis proposicionais: de 1 a 3 variáveis proposicionais

Processamento:

1. Identificar se a fórmula tem sintaxe correta
2. Se fórmula correta
 - a. Traduzir a fórmula (na String) na sua correspondente em linguagem de programação
 - b. Se fórmula correta, apresentar a **tabela verdade** para todas as interpretações possíveis;
 - c. Se fórmula incorreta, avisar que existe erro e não apresentar tabela

| Padrão para Fórmulas Bem Formadas (FBFs) | | | | | | | | | | | |
|------------------------------------------|------------------|---|------|------|------|------|------|---|---|---------------------------|------------------|
| # | Posição no Array | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Número Operações Binárias | Número Variáveis |
| | Tamanho String | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | |
| 1 | 1 | p | | | | | | | | 0 | 1 |
| 2 | 2 | ~ | p | | | | | | | | |
| 3 | 3 | p | opr1 | q | | | | | | 1 | 2 |
| 4 | 4 | ~ | p | opr1 | q | | | | | | |
| 5 | | p | opr1 | ~ | q | | | | | | |
| 6 | 5 | ~ | p | opr1 | ~ | q | | | | 2 | 3 |
| 7 | | p | opr1 | q | opr2 | r | | | | | |
| 8 | 6 | ~ | p | opr1 | q | opr2 | r | | | | |
| 9 | | p | opr1 | ~ | q | opr2 | r | | | | |
| 10 | | p | opr1 | q | opr2 | ~ | r | | | | |
| 11 | 7 | ~ | p | opr1 | ~ | q | opr2 | r | | | |
| 12 | | ~ | p | opr1 | q | opr2 | ~ | r | | | |
| 13 | | p | opr1 | ~ | q | opr2 | ~ | r | | | |
| 14 | 8 | ~ | p | opr1 | ~ | q | opr2 | ~ | r | | |

1.3 Exemplos de fórmulas (Strings) admitidas

| Exemplos de Fórmulas Bem Formadas (FBFs) | | | | | | | | |
|------------------------------------------|-----------------|---|---|---|---|---|---|---|
| 1 | p | | | | | | | |
| 2 | ~ p | | | | | | | |
| 3 | p v q | | | | | | | |
| 4 | ~ p v q | | | | | | | |
| 5 | p v ~ q | | | | | | | |
| 6 | ~ p v ~ q | | | | | | | |
| 7 | p v q ^ r | | | | | | | |
| 8 | ~ p v q ^ r | | | | | | | |
| 9 | p v ~ q ^ r | | | | | | | |
| 10 | p v q ^ ~ r | | | | | | | |
| 11 | ~ p v ~ q ^ r | | | | | | | |
| 12 | ~ p v q ^ ~ r | | | | | | | |
| 13 | p v ~ q ^ ~ r | | | | | | | |
| 14 | ~ p v ~ q ^ ~ r | | | | | | | |
| 15 | p | | | | | | | |
| 16 | ~ p | | | | | | | |
| 17 | p ^ q | | | | | | | |
| 18 | ~ p ^ q | | | | | | | |
| 19 | p ^ ~ q | | | | | | | |
| 20 | ~ p ^ ~ q | | | | | | | |
| 21 | p ^ q v r | | | | | | | |
| 22 | ~ p ^ q v r | | | | | | | |
| 23 | p ^ ~ q v r | | | | | | | |
| 24 | p ^ q v ~ r | | | | | | | |
| 25 | ~ p ^ ~ q v r | | | | | | | |
| 26 | ~ p ^ q v ~ r | | | | | | | |
| 27 | p ^ ~ q v ~ r | | | | | | | |
| 28 | ~ p ^ ~ q v ~ r | | | | | | | |
| 29 | ~ p ^ q v ~ r | | | | | | | |
| 30 | ~ p v ~ q v ~ r | | | | | | | |
| 31 | p v q v ~ r | | | | | | | |
| 32 | p ^ q ^ r | | | | | | | |
| índice | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

2 Ajuda para confecção do Programa em Java

2.1 Exemplo de código - Tabela Verdade com 2 variáveis proposicionais

```
public class Tabela2 {
    public static char printBit(boolean p) {
        if (p)
            return 'V';
        else
            return 'F';
    }

    public static void main (String[] args) {
        int i;
        boolean p, q;

        System.out.println("+---+---+-----+");
        System.out.println("| p | q | p -> q |");
        System.out.println("+---+---+-----+");

        for (i = 3; i >= 0; i--) {
            switch (i) {
                case 3:
                    p = true; q = true;
                    break;
                case 2:
                    p = true; q = false;
                    break;
                case 1:
                    p = false; q = true;
                    break;
                default:
                    p = false; q = false;
            }
            System.out.print ("| " + printBit(p) + " | " +
printBit(q) );
            System.out.println(" | " + printBit(!p || q ) + "
|");
        }
        System.out.println("+---+---+-----+");
    }
}
```

2.2 Exemplo de código - Tabela Verdade com 3 variáveis proposicionais

```
public class Tabela1 {
    public static char printBit(boolean p) {
        if (p)
            return 'V';
        else
            return 'F';
    }

    public static void main (String[] args) {
        int i;
        boolean p, q, r;

        System.out.println("+---+---+---+-----+");
        System.out.println("| p | q | r | p v q v r |");
        System.out.println("+---+---+---+-----+");

        for (i = 7; i >= 0; i--) {
            switch (i) {
                case 7:
                    p = true; q = true; r = true;
                    break;
                case 6:
                    p = true; q = true; r = false;
                    break;
                case 5:
                    p = true; q = false; r = true;
                    break;
                case 4:
                    p = true; q = false; r = false;
                    break;
                case 3:
                    p = false; q = true; r = true;
                    break;
                case 2:
                    p = false; q = true; r = false;
                    break;
                case 1:
                    p = false; q = false; r = true;
                    break;
                default:
                    p = false; q = false; r = false;
            }
            System.out.print ("| " + printBit(p) + " | " + printBit(q)
+ " | " + printBit(r) );
            System.out.println(" |      " + printBit(p || q || r) + "
|");
        }
        System.out.println("+---+---+---+-----+");
    }
}
```