

Relatório: Cálculo de Tabela-Verdade

Alunos: Fábio Melo e Luigge Lena

A Fórmula escolhida para utilização no Projeto:

$$H = (P \vee R) \wedge \neg S \rightarrow (P \wedge Q)$$

Para que a formula seja interpretada pelo **primeiro programa**, é feitas as seguintes modificações:

- a adição de uma quarta constante (S) e modificação das definições do índice VARS.

```
// numero de variaveis proposicionais na formula
#define VARS          4

// constantes para os indices das variaveis
#define P              0
#define Q              1
#define R              2
#define S              3
//adicionado variavel S.
```

- Alteração do retorno da função valor_formula() e o uso de dois registros temporários (calculo1, calculo2) para que seja possível o cálculo da fórmula sem manipular a macro “IMP”

```
int valor_formula()
{
    // para que o codigo funcione, temos que calcular os termos primeiro //
    int calculo1 = I[P] || I[R] && !I[S];
    int calculo2 = I[P] || I[Q];
    return IMP(calculo1, calculo2);
}
```

- Tais modificações geram o resultado esperado na execução do programa:

```

Calculo de tabela-verdade

Formula:
H = ((P \ / R) /\ ~S) -> (P \ / Q)

P | Q | R | S | H
-----
F | F | F | F | T
F | F | F | T | T
F | F | T | F | F
F | F | T | T | T
F | T | F | F | T
F | T | F | T | T
F | T | T | F | T
F | T | T | T | T
T | F | F | F | T
T | F | F | T | T
T | F | T | F | T
T | F | T | T | T
T | T | F | F | T
T | T | F | T | T
T | T | T | F | T
T | T | T | T | T

Process returned 0 (0x0)   execution time : 0.095 s
Press any key to continue.

```

- Para o **Segundo Programa**, nós utilizamos de um método diferente. começando pela adição de funções, estruturas e definições para que seja possível o uso da variável S.

```

// tipo do no da arvore
typedef enum tagTipo {
    NEG, AND, OR, IMP, BIMP, P, Q, R, S
} Tipo;

// uma formula de logica

#define VARS      4
#define PIND      0
#define QIND      1
#define RIND      2
#define SIND      3

```

```

void inicializa_formula()
{
    nome[PIND] = 'P';
    nome[QIND] = 'Q';
    nome[RIND] = 'R';
    nome[SIND] = 'S';

    Formula* var_s()
    {
        return cria_formula(S, NULL, NULL);
    }

    case A:
    case S:
        return I[indice_variavel(f->tipo)];

```

- É montada a função main que retorna a tabela-verdade correta:

```

int main(int argc, char **argv)
{
    // ((P \ / R) /\ ~S) -> (P \ / Q)
    Formula *f =

        imp(var_p(), var_q());
        imp(and(neg(var_s()), or(var_p(), var_r())), or(var_p(), var_q()));

    printf("Calculo de tabela-verdade\n\n");

    mostra_tabela(f);

    destroi_formula(f);

    return 0;
}

```

- Execução do Programa:

```
Calculo de tabela-verdade

Formula:
H = ((P \ / R) /\ ~S) -> (P \ / Q)

P | Q | R | S | H
-----
F | F | F | F | T
F | F | F | T | T
F | F | T | F | T
F | F | T | T | T
F | T | F | F | T
F | T | F | T | T
F | T | T | F | T
F | T | T | T | T
T | F | F | F | F
T | F | F | T | F
T | F | T | F | F
T | F | T | T | F
T | T | F | F | T
T | T | F | T | T
T | T | T | F | T
T | T | T | T | T

Process returned 0 (0x0)   execution time : 0.088 s
Press any key to continue.
```

- O código-fonte de ambos os programas modificados está incluso na pasta deste relatório.