

## Exercícios complementares - Operadores

1. Escreva um programa que receba um valor em decimal e converta para binário
2. Um programa que faça o oposto do exercício anterior, receba um número em binário e converta para decimal.
3. Faça a função `qtdeDeBitsParesLigados(int n)` que retorna o número de bits pares que estão ligados.<sup>1</sup>

Ex:

```
main() {
    qtdeDeBitsParesLigados(73); //se 73 = 0100 1001, então
    qtdeDeBitsParesLigados == 2
}
```

4. Escreva uma função `criptografa(int n)` que recebe um inteiro `n` com 8 bits (índices: 7,6,5,4,3,2,1,0) e que retorna esse inteiro embaralhando esses bits para a seguinte sequência (7,5,3,1,6,4,2,0)<sup>1</sup>

Ex:

```
main() {
    criptografa(73); // se 73 = 0100 1001, então criptografa(73) == 0010
    1001 == 41
}
```

5. Faça uma função `pisca(int milissegundos)` que ora liga todos os bits pares e ora liga todos os bits ímpares num intervalo de 2 segundo entre as mudanças, imitando o efeito de um pisca-pisca de natal. Apresente os valores em binários de como ficaria a saída. O valor de milissegundos é usado para fazer uma espera entre a alternância, e para isso use a função `Sleep(<int milissegundos>)` da biblioteca `windows.h`.<sup>1</sup>
6. Faça uma função `piscaUmIndoEVoltando(int milissegundos)` que faz os leds pares serem ligados sequencialmente, um de cada vez e de forma crescente, e em seguida, os leds ímpares serem ligados de forma decrescente.<sup>1</sup>
7. Faça um programa que leia um byte do teclado e a seguir zere os bits 3 e 4, e inverta os bits 0 e 7. O resultado deverá ser mostrado em hexadecimal na tela.<sup>2</sup>

---

<sup>1</sup> Ensino Superior e Técnico em Informática (UNIBRATEC) - Professor: Frederico Brito Fernandes

<sup>2</sup> Operadores Bit-A-Bit - [mesquita@cefetsp.br](mailto:mesquita@cefetsp.br)