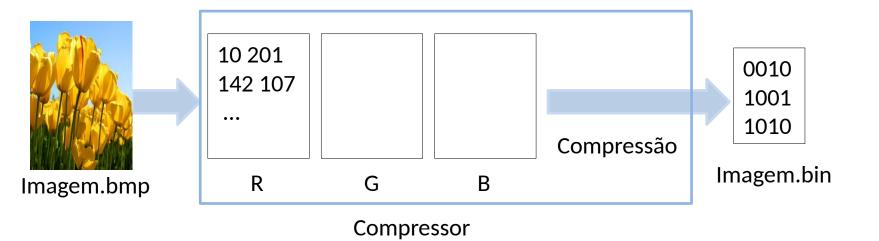
Como Gravar Bits (apenas) em arquivos



 Compressão implica em converter um conjunto de símbolos de entrada em um conjunto de códigos de menor tamanho como saída.

- O quê usar para manipular/armazenar os códigos?
 - "Uma E.D."!
- Como representar os códigos?
 - Int/char/float/...?

Exemplo:

 Um vetor de structs, onde cada struct contém um uchar (int) e uma vetor de char (string).

10	['0', '1', '\0']
201	['0', '0', '\0']
142	['1', '0', '0', '\0']
107	['1', '1', '\0']

- uchar: 1 byte

- string: 1 byte para cada posição!

- Como irá gravar em arquivo?
 - Deveria gravar apenas os bits de cada símbolo!

Ideia:

- Usar um buffer de 8 bits (1 byte = uchar, p.e)
- Quando o buffer encher, grava-se em arquivo.

10	['0', '1', '\0']
201	['0', '0', '\0']
142	['1', '0', '0', '\0']
107	['1', '1', '\0']
107	['1', '1', '\0']
142	['1', '0', '0', '\0']
•••	



Próximo buffer

<u>i=0</u> Buffer: 00000000 << 1: 00000000 : 0 Buffer: 00000000	i=3 Buffer: 00000010 << 1: 00000100 : 0 Buffer: 00000100	<u>i=6</u> Buffer: 00010010 << 1: 00100100 : 0 Buffer: 00100100
<u>i=1</u> Buffer: 00000000 << 1: 00000000 : 1 Buffer: 00000001	<u>i=4</u> Buffer: 00000100 << 1: 00001000 : 1 Buffer: 00001001	<u>i=7</u> Buffer: 00100100 << 1: 01001000 : 1 Buffer: 01001001
<u>i=2</u> Buffer: 00000001 << 1: 00000010 : 0 Buffer: 00000010	<u>i=5</u> Buffer: 00001001 << 1: 00010010 : 0 Buffer: 00010010	