



Apresente o resultado da seguinte operação aritmética:

$$323_4 + 010011_2$$

- na base hexadecimal:
- converta para a base dez:

Teste-ensaio (2)



Considere o estado parcial de um PC com um processador IA-32 (*little endian*), ilustrado em baixo, com o excerto final do código de uma função em execução e uma visão do quadro de ativação dessa função na *stack*.

Registos	Memória (código)	Memória (dados)
%eip = 0x00004050	0x4050 popl %eax	0x06ffc: 0x50 0x00 0x00 0x00
%ebp = 0x00007000	0x4051 addl 12(%ebp),%eax	0x07000: 0x10 0x70 0x00 0x00
%eax = 0x00002300	0x4054 pushl %eax	0x07004: 0xb5 0x40 0x00 0x00
%ecx = 0x00000011	0x4055 leave	0x07008: 0x20 0x00 0x11 0x00
%esp = 0x00006ffc	0x4056 ret	0x0700c: 0x01 0x00 0xff 0x00

- **indique** a mnemónica em *assembly* da próxima instrução a ser executada (sem especificar o operando)
- **apresente** o conteúdo do registo `%eax` em hexadecimal, no final da execução desta função

Teste-ensaio (3)



Considere a adição de 2 inteiros, `addw %ax, %bx`, na arquitetura de um processador IA-16, em que os inteiros são representados em complemento para 2:

- sabendo que o conteúdo desses registos antes da adição é `0x204` (em `%ax`) e `0xc` (em `%bx`), **mostre** o conteúdo em decimal dos registos `%ax` e `%bx` no final da execução desta soma
- sabendo que o conteúdo desses registos antes da adição é `0x1e` (em `%ax`) e `0x1f2` (em `%bx`), **mostre** o conteúdo em decimal dos registos `%ax` e `%bx` no final da execução desta soma

Teste-ensaio (4)



Considere a arquitetura de um processador IA-16, que suporta uma representação de números reais usando uma versão reduzida da norma IEEE 754 com 16 bits: 5 bits para o expoente, 10 para a mantissa e 1 para o sinal; não esquecer os casos de exceção:

- **mostre, em binário**, o valor que se encontra armazenado num destes registos FP de 16 bits, se o seu conteúdo tiver o valor $0.12625 * 10^2$
- **mostre, em decimal**, que valor que se encontra armazenado num destes registos FP de 16 bits, se o seu conteúdo for $0x5648$

Teste-ensaio (5)



O estado (parcial) de um sistema com um processador IA-16 (*little endian*) pode ser descrito por:

```
ip = 0x3050   %sp = 0x0f46   %ax = 0x0010   %bx = 0x0100
```

Conteúdo de 8 células de memória, em hexadecimal, a partir de cada um dos endereços:

```
0x0108: 05  00  ef  ff  01  1a  24  1f
0x0f42: 1a  02  5f  17  ab  23  7f  2b
```

Considere a execução da seguinte instrução, que já foi decodificada: `addw %ax, 0x0a(%bx)`

Mostre o 1º bloco de informação que circula em cada um dos seguintes barramentos, em hexadecimal:

- **de endereços**
- **de dados**