

### Trabalho Prático 2 - Simulador AHMES

Escrever um programa para o simulador Ahmes que realize a conversão de números sem sinal (somente positivos) de formato inteiro para ponto flutuante, e vice-versa (de ponto flutuante para inteiro).

O número inteiro é representado em 8 bits, sem sinal (números positivos, entre 0 e 255). O número em ponto flutuante é representado em 16 bits, sendo 8 bits para mantissa (somente positiva, normalizada, com a vírgula localizada à esquerda do bit mais significativo) e 8 bits para o expoente (em complemento de 2, entre -128 e +127).

Para este formato em ponto flutuante, valem as seguintes representações:

- O zero é representado por mantissa zero e expoente -128
- O maior positivo é representado por mantissa 255 e expoente 127
- O menor positivo (não zero) é representado por mantissa 128 e expoente -128

Para a definição da conversão devem ser obrigatoriamente utilizadas as seguintes posições:

Palavra 128 – número inteiro

Palavra 129 – expoente do número em ponto flutuante

Palavra 130 – mantissa do número em ponto flutuante

Palavra 131 – indicador de conversão (0: conversão de inteiro para ponto flutuante;  
1: conversão de ponto flutuante para inteiro)

No formato definido acima, a conversão de inteiro para ponto flutuante sempre é possível, de forma exata e sem erro. Na conversão de ponto flutuante para inteiro, as seguintes convenções devem ser utilizadas:

- Se a mantissa for zero, o resultado é zero, independente do valor do expoente.
- Valores em ponto flutuante menores que 0,5 são convertidos para zero.
- Valores em ponto flutuante maiores que 255 são convertidos para 255.
- Valores de fração menores que 0,5 arredondam o valor do número inteiro para baixo (na direção do zero), e valores de fração maiores ou iguais a 0,5 arredondam o valor do número inteiro para cima (na direção do 255).

Os trabalhos serão corrigidos de forma automática, com **20** valores diferentes. Portanto, devem ser observadas rigorosamente as seguintes especificações:

- o código do programa deve iniciar no endereço 0 da memória.
- a primeira instrução executável deve estar no endereço 0.
- os endereços dos bytes dos números inteiro e ponto flutuante devem ser exatamente os especificados acima, inclusive na ordem dos bytes.
- usar para variáveis adicionais os endereços de memória de 132 em diante.
- na conversão de inteiro para ponto flutuante, o valor do endereço 128 não deve ser modificado, e na conversão de ponto flutuante para inteiro os valores dos endereços 129 e 130 não devem ser modificados.
- o número em ponto flutuante fornecido como entrada segue a descrição acima, e está garantidamente normalizado.

O programa deverá ser desenvolvido com o uso do montador Daedalus, sendo a documentação do trabalho incluída como comentário no código fonte (não fazer relatório à parte). O programa executável em memória deverá ser entregue no Moodle, na área de “Entrega do Segundo Trabalho”, juntamente com uma listagem comentada do programa fonte no Daedalus. Para os nomes dos arquivos, utilize a letra inicial do seu primeiro nome, seguida do seu número de identificação. Assim, por exemplo, o aluno João José da Silva, cartão 123456, deve denominar o seu arquivo de J123456.MEM e J123456.AHD. Como o Moodle só permite o envio de um único arquivo, compacte os arquivos usando Zip ou Rar.

**Data de Entrega: 23/06/2008 (Turma A)**

Exemplos de conversões de inteiro para ponto flutuante (valores em decimal)

Endereço	Entrada		Resultado	
	128	129	130	
Caso 1	0	-128	0	
Caso 2	1	1	128	
Caso 3	4	3	128	
Caso 4	7	3	224	
Caso 5	64	7	128	
Caso 6	127	7	254	
Caso 7	128	8	128	
Caso 8	255	8	255	

Exemplos de conversões de ponto flutuante para inteiro (valores em decimal)

Endereço	Entrada		Resultado	
	129	130	128	
Caso 1	qualquer	0	0	
Caso 2	-1	255	0	
Caso 3	-120	128	0	
Caso 4	0	128	1	
Caso 5	1	192	2	
Caso 6	1	255	2	
Caso 7	1	191	1	
Caso 8	9	128	255	
Caso 9	9	255	255	
Caso 10	120	128	255	
Caso 11	8	156	156	