

Trabalho Prático 1 - Simulador NEANDER

Escrever um programa para o simulador Neander que implemente as quatro operações aritméticas básicas (soma, subtração, multiplicação e divisão) sobre números inteiros positivos, representados em complemento de dois (ou seja, os operados estão na faixa entre 0 e 127, inclusive).

Para a definição da operação devem ser obrigatoriamente utilizadas as seguintes posições de memória:

Palavra 128 – primeiro operando da operação (A)

Palavra 129 – segundo operando da operação (B)

Palavra 130 – código da operação:

1 – multiplicação de $A * B$

2 – divisão de A / B

3 – soma de $A + B$

4 – subtração de $A - B$

Palavra 131 – código de erro (veja abaixo)

Palavra 132 – resultado da operação

A operação de divisão corresponde a uma divisão inteira, ou seja, o quociente é um número inteiro e o resto é descartado. Não é necessário realizar quaisquer testes sobre a correção do resultado da operação, ou seja, eventuais "vai-um" e estouros de representação devem ser ignorados. A validade das entradas deve ser indicada através dos seguintes códigos na posição 131 de memória:

Código 0: entradas corretas

Código 1: primeiro operando fora da faixa, ou seja, negativo

Código 2: segundo operando fora da faixa, ou seja, negativo

Código 3: ambos os operandos fora da faixa

Código 4: operandos dentro da faixa, mas código de operação inválido

Código 5: tentativa de divisão por zero

Se os operandos forem inválidos, não é necessário testar a validade do código da operação. Em caso de qualquer erro, deve ser devolvido zero como resultado da operação.

Os valores das posições de memória 128 a 130 não devem ser alterados pelo programa.

Dicas:

1. O simulador Neander não possui operação de subtração. Entretanto, uma subtração pode ser transformada em uma soma através do complemento do subtraendo.
2. O simulador Neander não possui operação de multiplicação. Entretanto, uma multiplicação pode ser transformada em uma série de somas sucessivas.
3. O simulador Neander não possui operação de divisão. Entretanto, o quociente de uma divisão pode ser calculado através de uma série de subtrações sucessivas.

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

- o código do programa deve iniciar na posição 0 da memória.
- a primeira instrução executável deve estar na posição 0 da memória.
- os endereços das variáveis de entrada devem ser exatamente os especificados acima.
- para constantes e variáveis adicionais, ou para trechos extras de programa, usar as posições de memória de 133 em diante.
- o programa será executado 20 vezes de forma consecutiva (sem ser carregado de disco a cada vez); portanto é necessário inicializar todas as variáveis utilizadas.

O trabalho deverá ser entregue no Moodle, na área de “Entrega do Primeiro Trabalho”, na forma de um arquivo compactado (formato Zip ou Rar) composto por:

- um arquivo de memória do Neander, contendo o programa.
- um arquivo texto, com documentação contendo uma breve descrição do método utilizado e uma listagem do programa fonte usando mnemônicos simbólicos para as instruções (LDA, STA, etc ...) e indicando em decimal os endereços de memória nos quais as instruções ficam armazenadas e os endereços dos operandos. Não se esqueça de incluir seu nome completo e seu número de cartão nas primeiras linhas deste arquivo.

Para o nome dos arquivos (memória, texto e compactado), utilize a letra inicial do seu primeiro nome, seguida do seu número de cartão, com 6 dígitos. Assim, por exemplo, o aluno João José da Silva, cartão número 123456, deve denominar os seus arquivos de **J123456.MEM**, **J123456.TXT** e **J123456.ZIP** (ou **RAR**).

Data de Entrega: 30/04/2009 via http://moodle.inf.ufrgs.br
--

Exemplos de casos de teste (todos os valores estão indicados no sistema decimal)

Endereço	128	129	130	131	132
Caso 1	128	8	2	1	0
Caso 2	44	1	5	4	0
Caso 3	29	1	3	0	30
Caso 4	115	15	3	0	130
Caso 5	105	127	4	0	234
Caso 6	47	3	1	0	141
Caso 7	122	100	2	0	1
Caso 8	28	0	2	5	0

Endereço	128	129	130	131	132
Caso 9	18	12	3	0	30
Caso 10	105	56	4	0	49
Caso 11	12	12	1	0	144
Caso 12	103	2	2	0	51
Caso 13	101	5	2	0	20
Caso 14	45	130	5	2	0
Caso 15	128	129	3	3	0
Caso 16	35	32	1	0	96