

TRABALHO PRÁTICO 2 – LIGADOR

Deiziane Silva

Lorena Barreto

1. Descrição Geral

Nesse trabalho prático foi desenvolvido um ligador que atua em conjunto com o montador desenvolvido previamente no trabalho anterior. O montador foi modificado de forma com que esse produza dois arquivos de saída, um com as instruções montadas e sua tabela de símbolos. O ligador tem como entrada esses dois arquivos, e após executar todos os procedimentos de ligação, este gera dois arquivos: `saida.o` e `saida.mif`. O `saida.o` serve apenas como auxiliar na geração do `saida.mif`, que é o arquivo de saída obrigatório utilizado para testar o ligador no CPUSim.

2. Decisões de Projeto

Foi decidido que o montador geraria dois arquivos para cada modulo do programa, um com as instruções já montadas e outro que contém sua tabela de símbolos. O formato de saída escolhido por nós foi o `.o` que é a extensão de arquivos objeto que são gerados pelo montador. O montador lê os arquivos de entrada da pasta `tst` e gera os arquivos objeto também nesta pasta com a extensão `.o`. Da mesma forma, foi decidido que o ligador geraria um arquivo `saida.o` somente para auxiliar na criação do arquivo `.mif`, estes arquivos se encontram também na pasta `tst`.

Também assumimos como decisão de projeto que somente o arquivo `main` teria pseudo-instruções `.extern`. Essa decisão foi tomada para fins de simplificação do programa.

Ao executar o programa, os argumentos que devem ser passados seguem o formato obrigatório informado na documentação. Porém, apesar do montador gerar dois arquivos, um com o código e o outro com a tabela de símbolos, apenas o primeiro deve ser chamado, não sendo necessário chamar também sua tabela de símbolos pois isso já é devidamente tratado no código da implementação do ligador. Para executar o programa obrigatório, por exemplo, deve-se fazer a chamada da seguinte forma:

```
./ligador saída.mif main.o major.o minor.o sum.o multiply.o mean.o
```

Eles serão lidos utilizando-se o vetor de argumentos do `main argv []`.

Note que, nesse caso, o montador deve ser previamente executado 6 vezes, uma para o `main`, e uma para cada módulo.

3. Testes

Para fazer os testes foram utilizados dois programas, um é o arquivo mostrado na especificação, que realiza a multiplicação de dois números e imprime o resultado. O outro é o programa teste obrigatório. Este programa lê quatro números, sendo o primeiro o código da operação a ser realizada (de 1 a 5), os outros três são os números que serão manipulados dependendo da operação escolhida. As operações são:

- 1 – Apresenta o maior dos 3 números
- 2- Apresenta o menor dos 3 números
- 3 – Apresenta a soma dos 3 números
- 4 - Apresenta o produto dos 3 números
- 5 – Apresenta a média dos 3 números

Cada operação possui um módulo próprio que é chamado pelo main através da pseudo-instrução extern.

Execução no CPUSim:

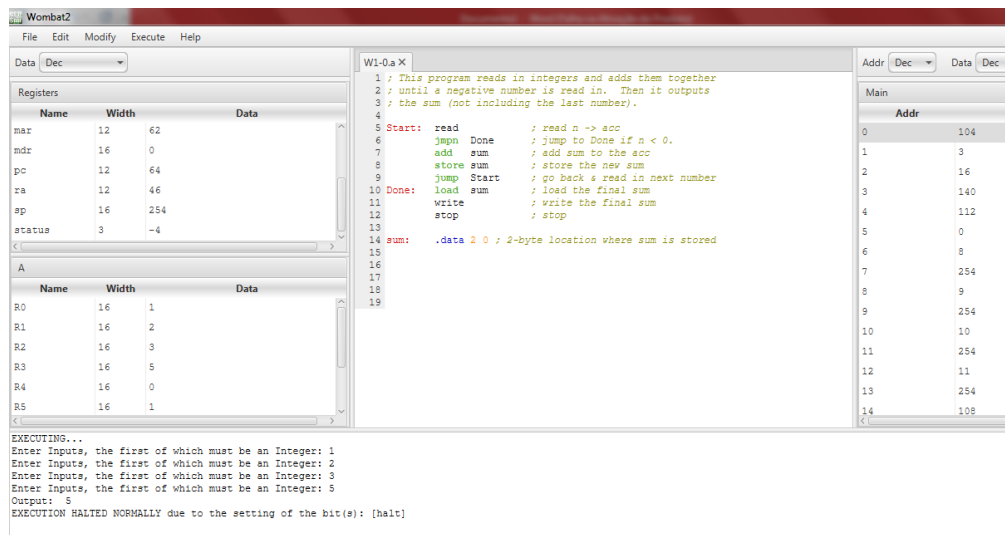


Figura 1 - Operação 1 (Maior dos 3)

Na figura 1, a operação escolhida foi a de apresentar o maior número sendo que o primeiro deles foi 2, o segundo 3 e o terceiro 5. O resultado apresentado pelo programa foi 5.

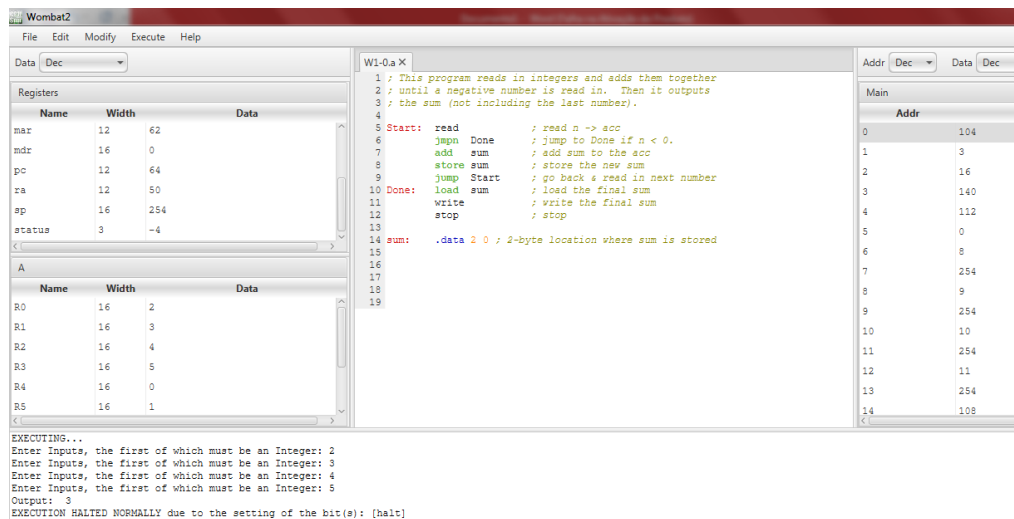


Figura 2- Operação 2 (Menor dos 3)

Na figura 2, a operação escolhida foi de apresentar o menor dos 3. O primeiro número digitado foi 3, o segundo 4 e o terceiro 5. O resultado foi 3.

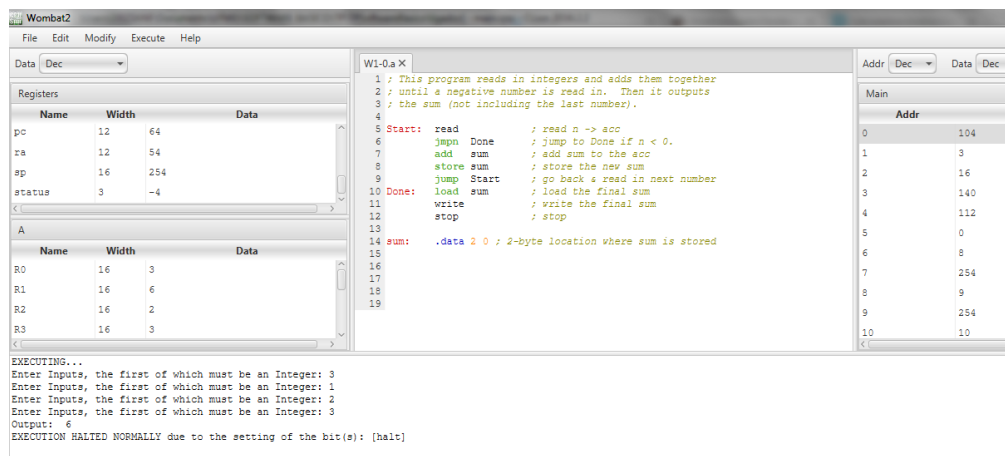


Figura 3 - Operação 3 (Soma dos 3)

Na figura 3, a operação escolhida foi a soma. Os números digitados foram 1, 2 e 3. O resultado foi 6.

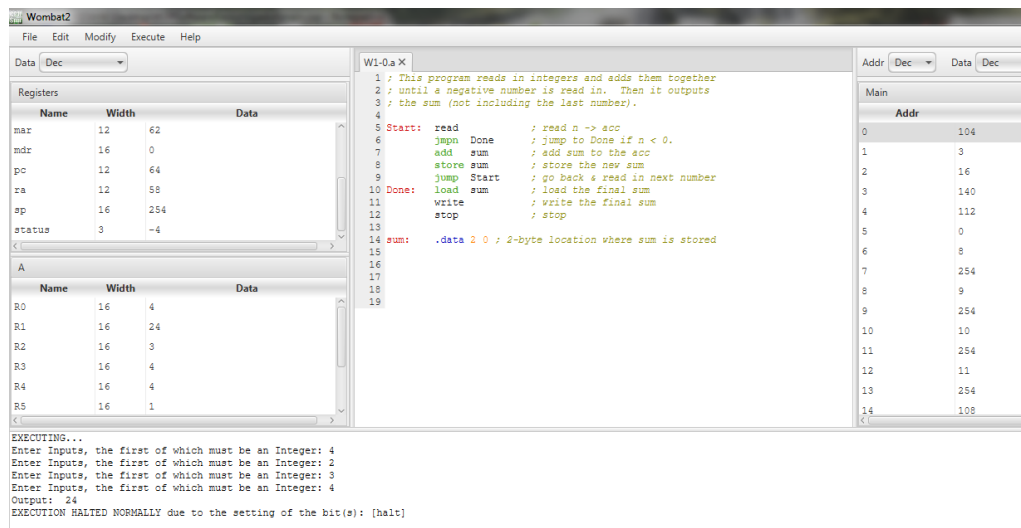


Figura 4 - Operação 4 (Produto dos 3)

Na figura 4, a operação escolhida foi o produto. Os números digitados foram 2, 3 e 4. O resultado foi 24.

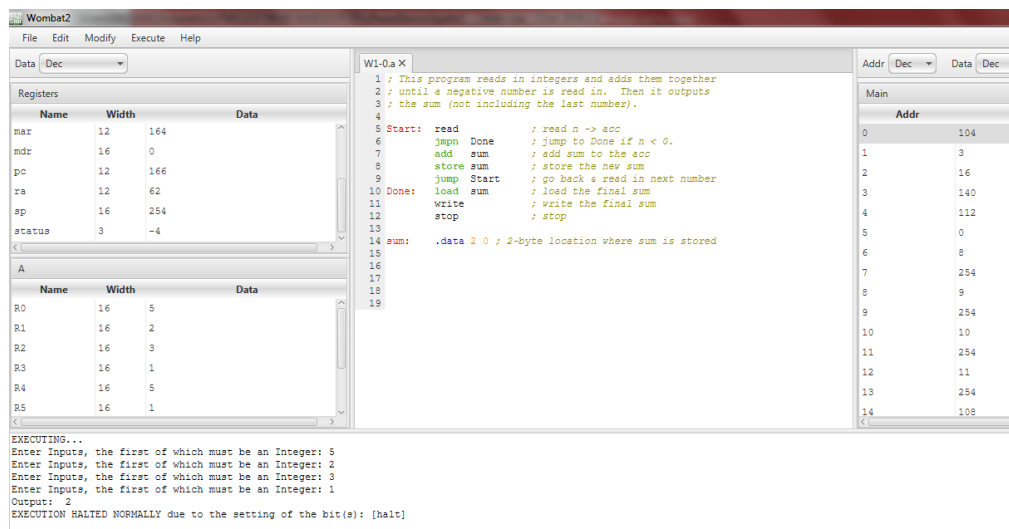


Figura 5 - Operação 5 (média dos 3)

Na figura 5, a operação escolhida foi a média. Os números digitados foram 2, 3 e 1. O resultado foi 2.

Os testes foram executados utilizando a máquina Wombat2 no CPUSim 4.0.9 no Windows 7.