

## EP – 1 – Documentação – Arquitetura de Computadores

### Grupo:

Guilherme Kenzo Silva Oshiro – N°USP 11314988

Guilherme Rodrigues Pisni - N° USP 11270851

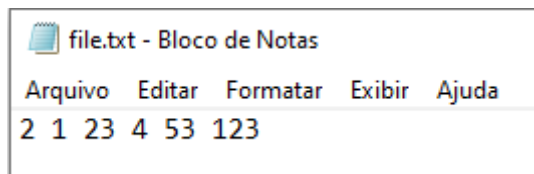
Leonardo Coppola Biazucci – N° USP 10723728

Mark Poll Herrmann – N° USP: 11208291

### Modo de uso:

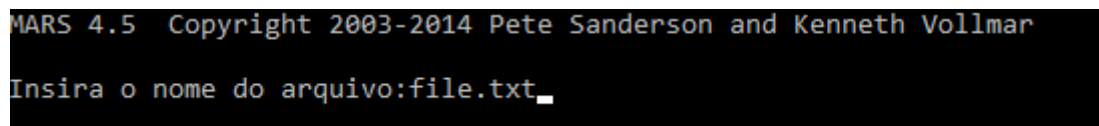
De modo a funcionar corretamente, o programa deve ser compilado e executado no MARS 4.5.

Ao executar o programa, será mostrada ao usuário uma mensagem perguntando o nome do arquivo onde estão armazenados os números para ordenação. Supondo que números estão armazenados em um arquivo chamado “file.txt” como demonstra a imagem abaixo.



**Obs: É NECESSÁRIO QUE OS NÚMEROS ESTEJAM SEPARADOS POR ESPAÇO NO ARQUIVO PARA O FUNCIONAMENTO CORRETO DO PROGRAMA.**

Assim que a mensagem for mostrada, o usuário deverá digitar o nome do arquivo (conforme a imagem abaixo) e apertar ENTER.

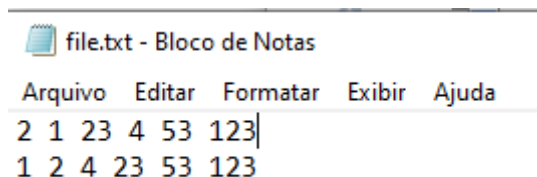


**Obs: É NECESSÁRIO QUE O ARQUIVO ESTEJA NO MESMO DIRETÓRIO DO MARS E DO ARQUIVO .ASM (Código-fonte)**

Então será mostrada uma mensagem perguntando ao usuário o tipo de algoritmo de ordenação desejado, ele então deverá digitar o número correspondente ao indicado na mensagem, sendo 0 para quickSort e 1 para selectionSort. Supondo que nesse caso o usuário queira utilizar o quickSort, ele deverá então digitar “0” e apertar ENTER para prosseguir.

```
MARS 4.5 Copyright 2003-2014 Pete Sanderson and Kenneth Vollmar
Insira o nome do arquivo:file.txt
Insira um inteiro correspondente ao tipo de algoritmo desejado (0 para quicksort e 1 para selectionsort): 0
```

Então o arquivo será atualizado, e no final do arquivo indicado estarão escritos, de forma ordenada, a sequência numérica que estava anteriormente no arquivo. Como mostra a imagem abaixo:



```
file.txt - Bloco de Notas
Arquivo  Editar  Formatar  Exibir  Ajuda
2 1 23 4 53 123
1 2 4 23 53 123
```

## Readfile:

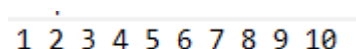
Função responsável pela leitura do arquivo desejado;

Esta função tem como finalidade, ler o arquivo e armazenar os números do arquivo no endereço indicado por “array”.

Para isto é necessário uma função que transforma a string em inteiros e assim partimos para a função “atoi”.

## atoi:

Função responsável por transformar strings em inteiros e para que funcione corretamente **É NECESSÁRIO QUE OS NÚMEROS ESTEJAM SEPARADOS POR ESPAÇO**. Desta forma:



```
1 2 3 4 5 6 7 8 9 10
```

## Ordena:

Função que recebe no registrador \$a0, o endereço do array, em \$a1 o tipo de algoritmo de ordenação, sendo de valor 0 para quickSort e 1 para selectionSort e em \$a2 recebe o tamanho do array. Esta função e as inclusas dentro dela, foram traduzidas diretamente de um algoritmo que cumpre a mesma função na linguagem C. Esta função retorna o tamanho do array em v1.

## Writefile:

Função responsável pela escrita do array já ordenado no arquivo desejado.

Esta função tem como principal atributo a função “itoa”

**itoa:**

função responsável por transformar os números armazenados no array para string, podendo assim serem escritos no arquivo destino.