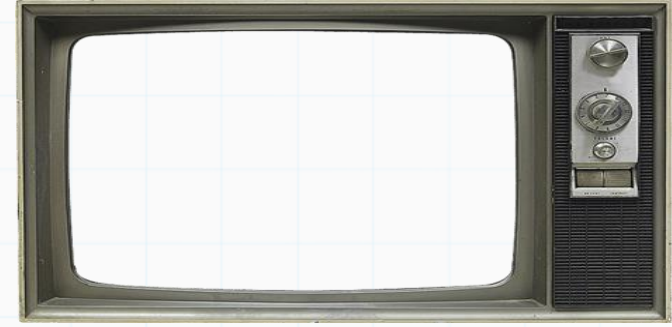


# Programação De Computadores

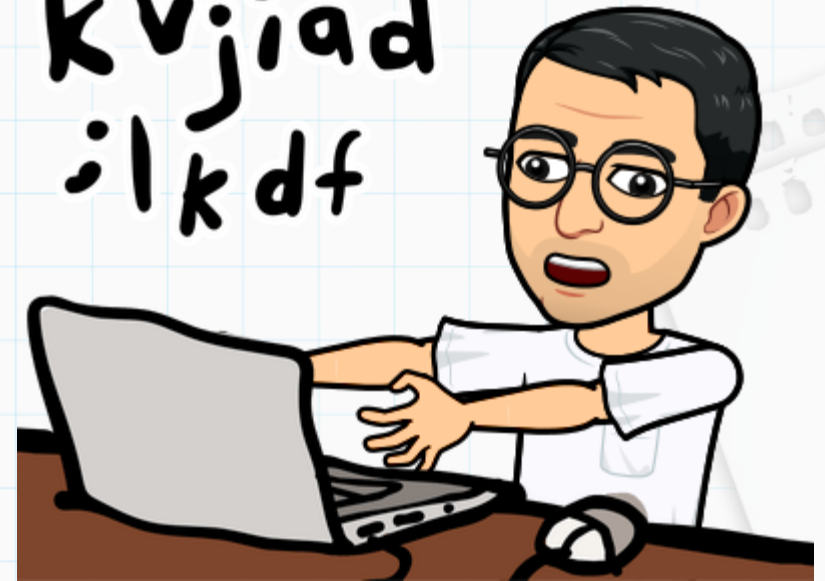
Professor : Yuri Frota

[www.ic.uff.br/~yuri/prog.html](http://www.ic.uff.br/~yuri/prog.html)

yuri@ic.uff.br



asdfghjkl;  
kvjiad  
ilkdf



# Arquivos- LAB



Exercício 1): Faça um programa que apure o resultado de uma votação para determinar o personagem do desenho “The Simpsons” favorito. Suponha que existam 5 candidatos cujos códigos de identificação são:

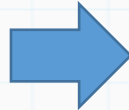
**1-Bart, 2-Homer, 3-Krusty, 4-Mr Burns, 5-Ned Flanders**

Considere um arquivo texto (denominado “votos.txt”) que contem, em cada linha, um determinado voto (um voto é representado pelo código de identificação do candidato). O programa deverá apresentar, como resultado, o nome do candidato e a quantidade de votos do candidato mais votado, o código de identificação e a quantidade de votos do candidato menos votado e a quantidade de votos nulos (um voto nulo é um voto cujo código de identificação é um inteiro diferente de 1,2,3,4,5).

votos - Bloco de Notas

Arquivo Editar Formatar E

2  
2  
4  
3  
1  
5  
4  
6  
4  
2  
8



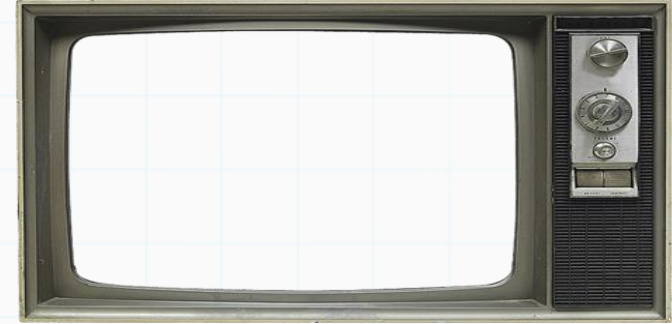
Ex. de execução:

Homer eleito com 6 votos!  
Candidato 1 foi o menos votado com 3 voto(s)!  
4 pessoas votaram nulo!



[código](#)

# Arquivos- LAB



Exercício 2): Um professor armazena em um arquivo texto “classe.txt” o número e o nome de cada aluno da turma da disciplina sob sua responsabilidade. Por questão de segurança, ele prefere armazenar as notas obtidas pelos alunos em cada prova em um outro arquivo texto (avalia.txt), onde cada linha contem o número do aluno e os valores das notas de 4 provas. Escreva um programa que permita consultar as notas de cada aluno a partir do seu nome. Seu programa deve receber o nome do aluno como entrada e buscar e imprimir a linha correspondente (com as notas) ao nome no arquivo “avalia.txt”, deve-se também imprimir a media das notas.

classe - Bloco de Notas				
Arquivo	Editar	Formatar	Exibir	
1	Quico			
2	Jaiminho			
3	Jirafales			
4	Nhonho			

avalia - Bloco de Notas				
Arquivo	Editar	Formatar	Exibir	
1	4.3	2.5	6.6	10.0
2	7.0	4.0	2.3	4.4
3	10.0	10.0	10.0	9.9
4	4.4	5.5	6.6	7.7

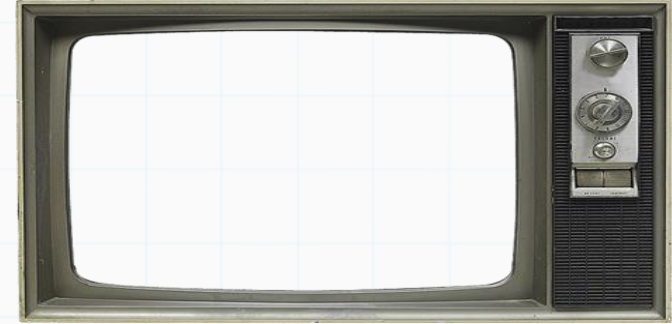
Ex. de execução:

Digite o nome do aluno: Nhonnho  
Aluno inexistente!

Digite o nome do aluno: Nhonho  
notas = 4.4 5.5 6.6 7.7  
media = 6.05

Digite o nome do aluno: Jirafales  
notas = 10.0 10.0 10.0 9.9  
media = 9.975

# Arquivos- LAB



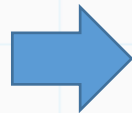
Exercício 3): Para uma possível futura viagem espacial, foram coletados e escritos em um arquivo texto os dados de várias pessoas. Nesse arquivo cada linha contém o primeiro nome da pessoa, seu último nome, seu ano de nascimento, seu desejo de ir para Marte ou para Júpiter, e seu peso na Terra (Veja exemplo).

Implemente um programa para ler os dados do arquivo texto (planetas.txt) e criar dois novos arquivos: o primeiro contendo o nome das pessoas com menos de 30 anos que desejam ir para Marte e o peso da pessoa em Marte; o segundo contendo o nome das pessoas com menos de 40 anos que desejam ir para Júpiter e o peso da pessoa em Júpiter. Sabe-se que uma pessoa com peso 100kg na Terra, pesa 38kg em Marte e 264kg em Júpiter.



planetas - Bloco de Notas

```
Arquivo  Editar  Formatar  Exibir  Ajuda
Maria Pacheco,1965,Marte,89
Goku Sayadim,1999,Marte,110
Pikachu Pokemon,1993,Jupiter,12
Sant Seya,1954,Jupiter,63
John Snow,1984,Marte,83
Rick Grimes,1976,Marte,93
Carl Grimes,1999,Jupiter,55
Lili Abitbol,2008,Marte,23
```



m - Bloco de Notas

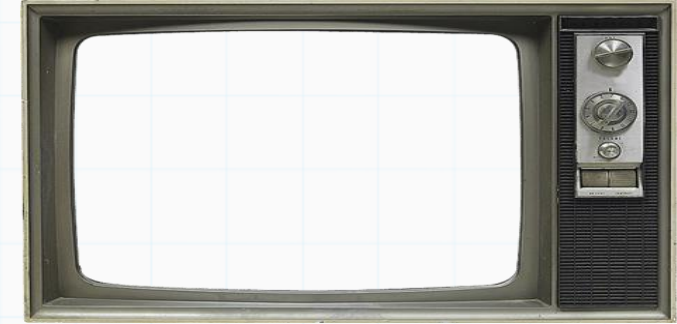
```
Arquivo  Editar  Formatar
Goku Sayadim 41.8
Lili Abitbol 8.74
```

j - Bloco de Notas

```
Arquivo  Editar  Formatar  Exib
Pikachu Pokemon 31.68
Carl Grimes 145.2
```



# Arquivos- LAB



Exercício 4): Implemente um programa que abre 2 arquivos (“nomes2.txt” e “notas2.txt”). O primeiro arquivo contem nomes e sobrenomes de alunos e o segundo arquivo contem as notas dos alunos. No primeiro arquivo, cada linha corresponde ao nome (e sobrenome) de um aluno e no segundo arquivo, cada linha corresponde a uma ou mais notas do aluno (separado por espaços), os dois arquivos tem o mesmo numero de linhas. Leia os dois arquivos e gere um terceiro (“medias2.txt”) que armazene o sobrenome (primeiro), nome e a média de cada aluno.

nomes2 - Bloco de Notas

```
Arquivo  Editar  Formatar  Exi
Joselino Barbacena
Sandoval Quaresma
Ze Bonitinho
Aldemar Vigario
Seu Boneco
```

notas2 - Bloco de Notas

```
Arquivo  Editar  Formatar  E
7.6
3.2 5.3 6.7
9.2 9.8
2.2 3.6
6.6 7.7 4.9 9.9
```



medias2 - Bloco de Notas

```
Arquivo  Editar  Formatar  Exibir  Ajuda
Barbacena Joselino 7.6
Quaresma Sandoval 5.0666666666666666
Bonitinho Ze 9.5
Vigario Aldemar 2.9000000000000004
Boneco Seu 7.275
```

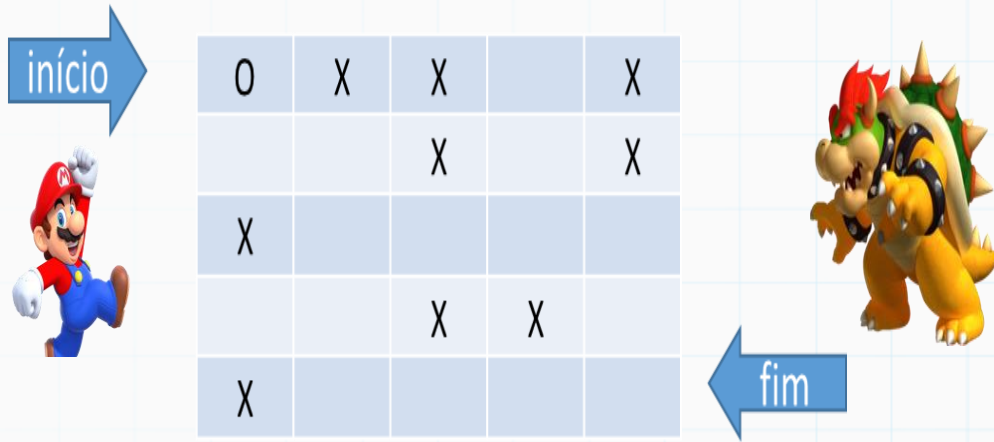


[código](#)



# Arquivos- LAB

Exercício 5) Labirinto: Dado uma matriz de caracteres de tamanho 5x5 descrita a seguir, onde "X" representa uma posição ocupada (parede ou bloco), " " representa uma posição não ocupada (livre), e "O" representa o avatar do jogador. O objetivo do jogo é levar o avatar da posição inicial (0,0), até a posição final (4,4).



Os movimentos do avatar são determinados por um arquivo “mov.txt” onde cada linha possui uma das quatro letras: E - esquerda , D - direita, C- cima, B-, baixo.

Ex:

B  
B  
D  
D  
B  
C



Continua...

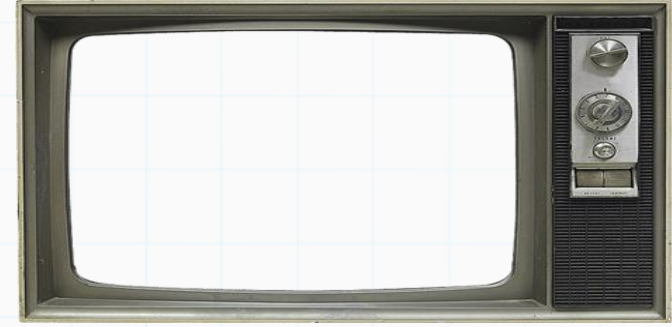


Faça um programa que iterativamente leia o arquivo de movimentos e realize o movimento na matriz. O programa deve imprimir a mensagem “vitória” ou “derrota” dependendo se o avatar conseguiu chegar na posição final (4,4) realizando os movimentos descritos no arquivo. Além disso, deve-se imprimir o número de **movimentos válidos** que o avatar realizou para chegar no final (posição final (4,4)).

# Arquivos- LAB

- Cuidado quando andar com o avatar para:

- não sair do labirinto,
- não andar sobre blocos ("X")
- e checar sempre a condição de vitória (chegar na posição 4,4).



Se o arquivo mandar realizar algum movimento não válido (fora do labirinto ou sobre um bloco), simplesmente passe para o próximo movimento.

Ex:

Para a sequência de movimentos apresentada no exemplo (B,B,D,D,B,C) temos :



# Arquivos- LAB

Para a sequência de movimentos apresentada no exemplo (B,B,D,D,B,C) temos :

O	X	X		X
		X		X
X				
		X	X	
X				



	X	X		X
O		X		X
X				
		X	X	
X				



	X	X		X
O		X		X
X				
		X	X	
X				

	X	X		X
	O	X		X
X				
		X	X	
X				



	X	X		X
	O	X		X
X				
		X	X	
X				



	X	X		X
		X		X
X	O			
		X	X	
X				

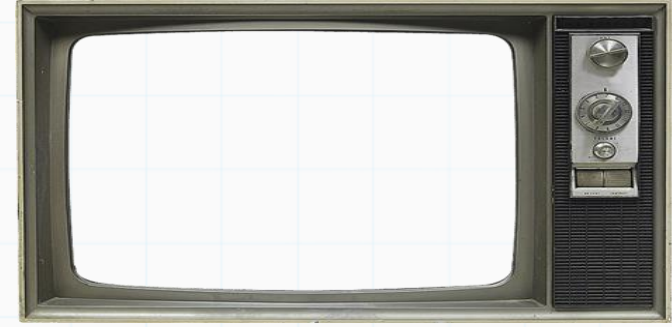
	X	X		X
	O	X		X
X				
		X	X	
X				

Continua...

Logo, a mensagem impressa seria “derrota”, com 4 movimentos válidos



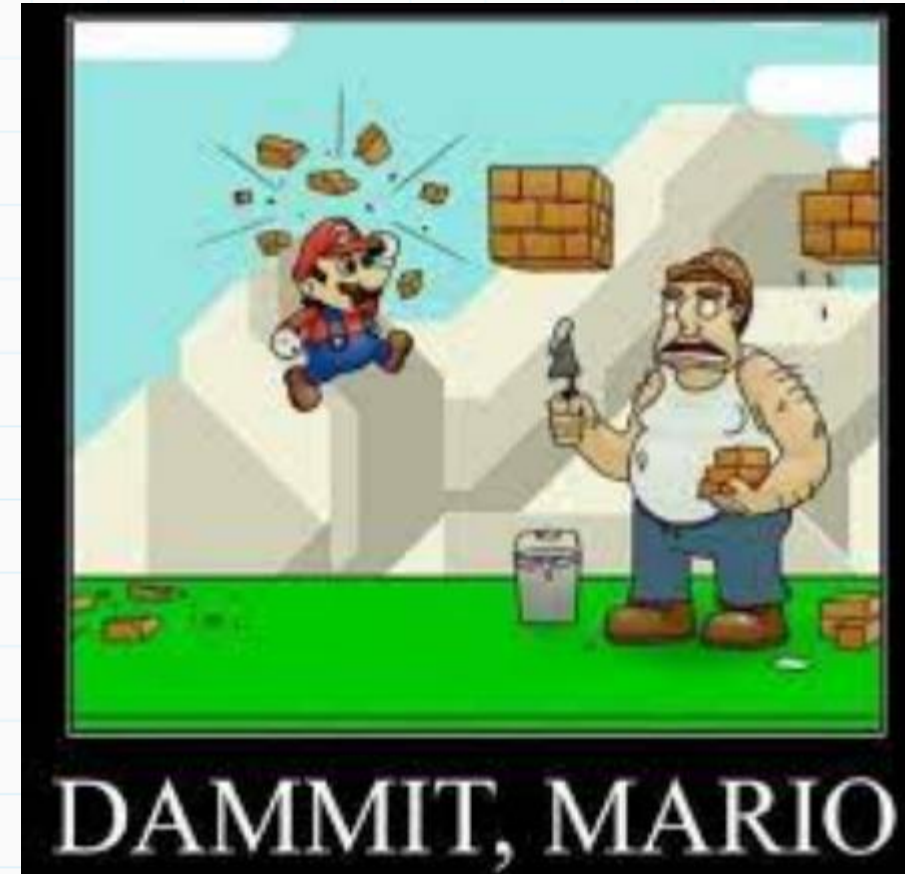
# Arquivos- LAB



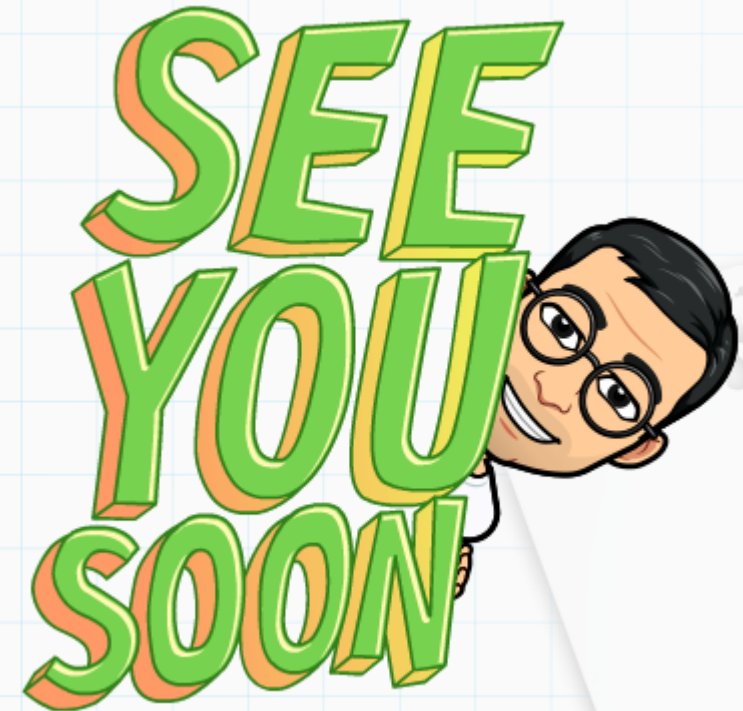
Dicas:

- 1) ao ler a letra representando o movimento, ela deve vir com quebra de linha “\n”. Use o comando strip() para deixar só a letra.
- 2) O jogo pode acabar antes do arquivo de movimentos chegar ao fim, se o avatar chegar a posição final, neste caso, nenhum movimento a mais deve ser realizado.
- 3) O arquivo labirinto.py servirá de base para seu programa, nele, a matriz (labirinto) já está definida.
- 4) Imprima a matriz a cada movimento realizado para acompanhar o percurso

Quais então seriam os resultados para as entradas mov1.txt, mov2.txt e mov3.txt ?



Até a próxima



Slides baseados no curso de Vanessa Braganholo