

ENSINO MÉDIO INTEGRADO - INFORMÁTICA

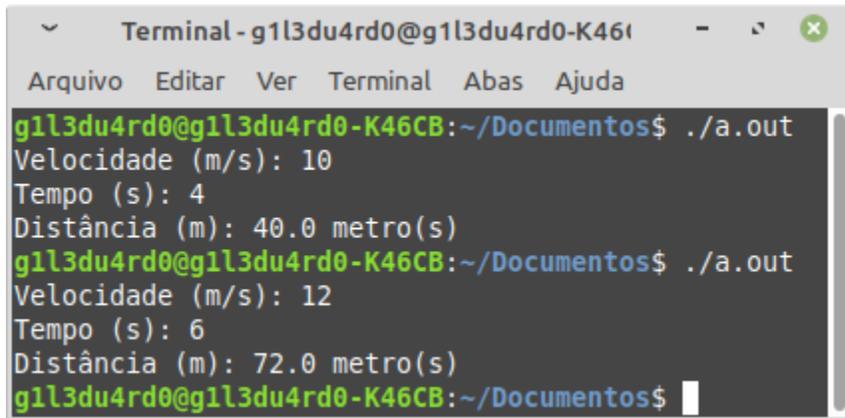
Disciplina de Linguagem de Programação

Prática 16: Lista Funções

Gil Eduardo de Andrade

EXERCÍCIOS RESOLVIDOS PELO PROFESSOR

1. Codifique uma função que calcule a distância percorrida por um veículo. A função deve receber como parâmetros a velocidade e o tempo total gasto para completar todo o percurso em questão. Veja o exemplo a seguir:



```
Terminal - g1l3du4rd0@g1l3du4rd0-K46C
Arquivo  Editar  Ver  Terminal  Abas  Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
Velocidade (m/s): 10
Tempo (s): 4
Distância (m): 40.0 metro(s)
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
Velocidade (m/s): 12
Tempo (s): 6
Distância (m): 72.0 metro(s)
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$
```

2. Codifique uma função que receba como parâmetros um vetor inteiro contendo de tamanho 10 (valores aleatórios 10 a 99) e uma variável inteira. Após a execução da função a variável inteira deve conter o valor do maior elemento do vetor, assim como todos os elementos do vetor devem ser substituídos pelo valor 0, exceto o maior valor. Veja o exemplo a seguir:

```
Terminal - g1l3du4rd0@g1l3du4rd0-K46(
Arquivo  Editar  Ver  Terminal  Abas  Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
Vetor: 34 18 69 91 20 16 53 90 18 82
Vetor: 00 00 00 91 00 00 00 00 00 00
Maior: 91
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
Vetor: 49 73 42 22 50 32 39 71 36 52
Vetor: 00 73 00 00 00 00 00 00 00 00
Maior: 73
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$
```

3. Crie uma matriz 5x5 (na main), preencha todo seu conteúdo com valores 0, exibindo-o. Após isso, codifique uma função para receber essa matriz (referência), alterando todo seu conteúdo para números aleatórios variando entre 10 e 99. Exiba o novo conteúdo da matriz (na main).

```
Terminal - g1l3du4rd0@g1l3du4rd0-K46(
Arquivo  Editar  Ver  Terminal  Abas  Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
Matriz:
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

Matriz:
25 45 13 19 17
81 73 98 76 34
54 80 16 38 75
97 26 13 33 50
52 56 17 28 32

g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$
```

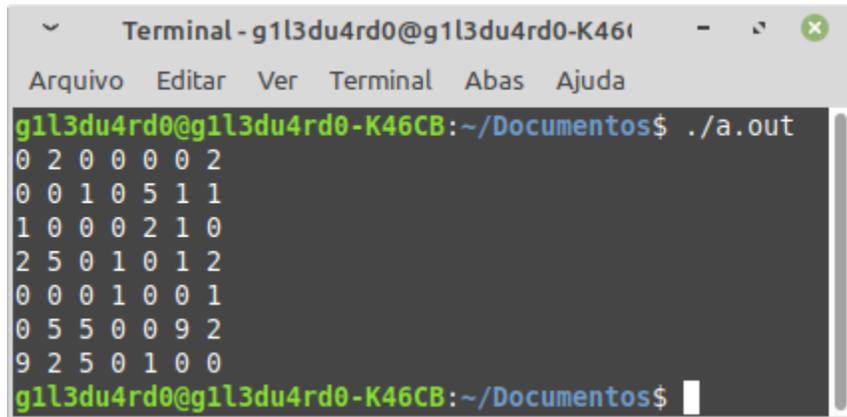
EXERCÍCIOS RESOLVIDOS PELO ALUNO - LABORATÓRIO

Entrega obrigatória até o término da aula

4. Codifique uma função chamada *montarCenario()*, que receba uma matriz inteira 7x7 e preencha-a aleatoriamente, seguindo o padrão:

| VALOR/PONTOS | QUANTIDADE |
|--------------|------------|
| 0 | 25 |
| 1 | 10 |
| 2 | 7 |
| 5 | 5 |
| 9 | 2 |

Após isso, exiba o seu conteúdo (na main). Veja o exemplo a seguir:



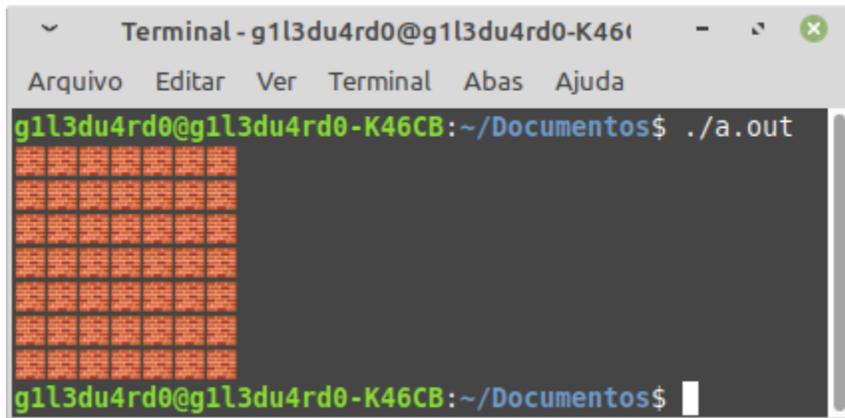
```
Terminal - g1l3du4rd0@g1l3du4rd0-K46CB
Arquivo Editar Ver Terminal Abas Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
0 2 0 0 0 0 2
0 0 1 0 5 1 1
1 0 0 0 2 1 0
2 5 0 1 0 1 2
0 0 0 1 0 0 1
0 5 5 0 0 9 2
9 2 5 0 1 0 0
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$
```

5. Codifique uma função chamada *exibirCenario()*, que receba uma matriz inteira 7x7 e um variável inteira flag, intérprete-a, e exiba seguindo o padrão:

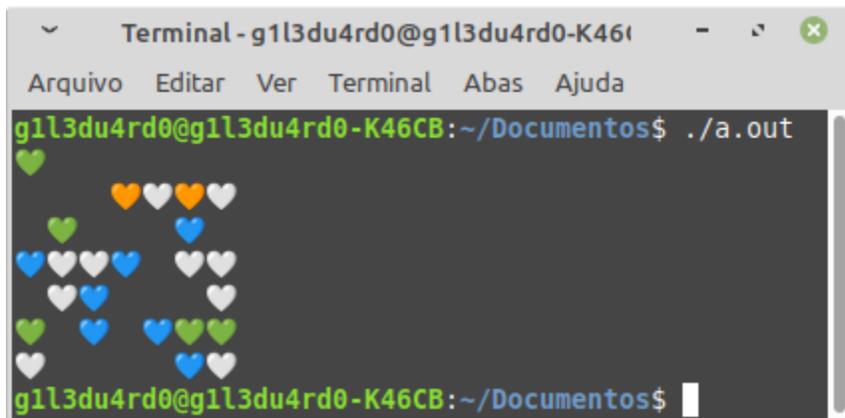
| VALOR/PONTOS | EMOJI |
|--------------|-------|
| < 10 | 🧱 |
| 11 | 💜 |
| 12 | 💙 |
| 15 | 💚 |

| | |
|----|---|
| 19 | ♥ |
| 10 | |

Caso o valor da variável inteira seja “0”, conforme a tabela anterior. Caso o valor seja “1”, os valores da matriz devem ser somados em 10 e exibidos conforme a tabela anterior. Veja o exemplo a seguir:



```
Terminal - g1l3du4rd0@g1l3du4rd0-K46CB
Arquivo Editar Ver Terminal Abas Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$
```

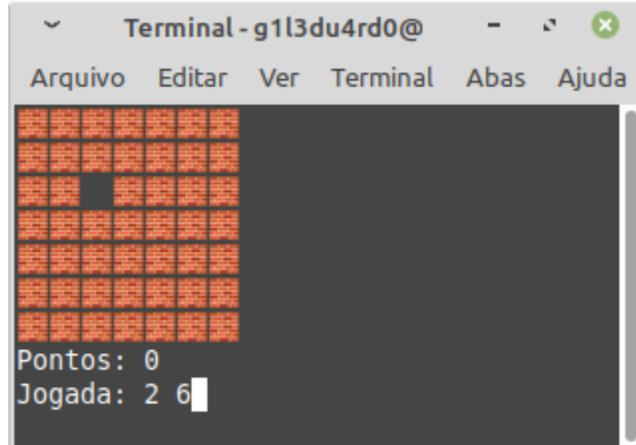
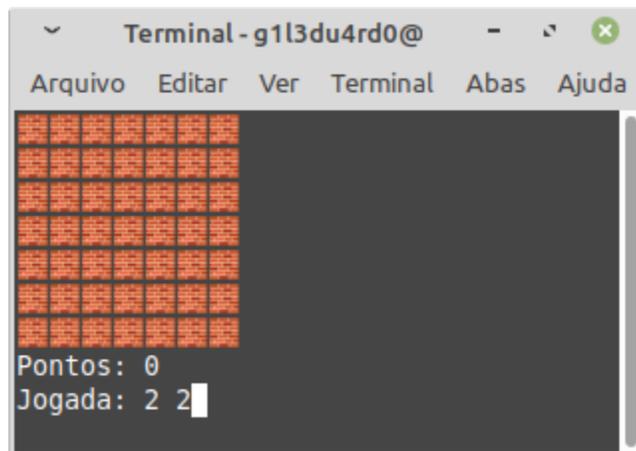


```
Terminal - g1l3du4rd0@g1l3du4rd0-K46CB
Arquivo Editar Ver Terminal Abas Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$
```

- Codifique uma função chamada *efetuarJogada()*, que receba uma matriz inteira 7x7 e dois valores inteiros, referentes a uma linha e uma coluna. A função deve retornar o valor inteiro contido na posição da matriz para linha e coluna e recebida. A função também deve atualizar a posição da matriz (linha, coluna) com seu valor somado a 10. Para testar a função criada, você deve invocar as funções nesta ordem:

```
montarCenario();
exibirCenario();
efetuarJogada();
exibirCenario();
```


7. Utilizando as funções criadas anteriormente construa um game onde o usuário deve jogar, continuamente, especificando uma linha e coluna, até atingir uma pontuação igual ou superior a 45 pontos. A cada jogada o cenário deve ser atualizado, removendo o emoji de tijolo e mostrando o que havia por baixo, assim como a pontuação do jogador. Ao final, todo o cenário deve ser exibido, através da remoção de todos os tijolos. Veja os exemplos abaixo:



```
Terminal - g1l3du4rd0@  
Arquivo Editar Ver Terminal Abas Ajuda  
Pontos: 5  
Jogada: 6 4
```

```
Terminal - g1l3du4rd0@  
Arquivo Editar Ver Terminal Abas Ajuda  
Pontos: 10  
Jogada: 0 5
```

```
Terminal - g1l3du4rd0@  
Arquivo Editar Ver Terminal Abas Ajuda  
Pontos: 12  
Jogada: 1 2
```

