



<http://www.nickgentry.com/>

Algoritmos e Programação de Computadores

Disciplina 113476

Prof. Alexandre Zaghetto
<http://alexandre.zaghetto.com>
zaghetto@unb.br

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Prática de Laboratório 06

Strings & Structs

1. Strings & Structs

1. Escreva um programa que solicita um nome ao usuário e imprime na tela do computador a mensagem "Olá, <nome_digitado>!" emoldurada em asteriscos. A largura da moldura deve adaptar-se automaticamente de acordo com o tamanho da *string*. Por exemplo, se o nome digitado for "Alexandre", o computador deve mostrar a seguinte mensagem na tela:

```
*****
*                               *
* Olá, Alexandre *
*                               *
*****
```

Se o nome tiver um comprimento diferente, como, por exemplo, "José" , a mensagem deve se adaptar automaticamente para:

```
*****
*                               *
* Olá, José *
*                               *
*****
```

1. Strings & Structs

2. Escreva um programa em linguagem C para criar um cadastro de vários alunos, incluindo a media como campo do cadastro. Para isto, defina um novo tipo de dado (*struct*) **taluno**, com os seguintes campos:

- `int nr`
- `char nome[MAX1]` (usando `define`, `MAX1 = 10`)
- `char departamento[MAX2]` (usando `define`, `MAX2 = 5`)
- `float nota[3]`
- `float media`

Declare um vetor `aluno[MAX]` do tipo `taluno`, com `define MAX 3`.

1. Strings & Structs

Percorra **registro a registro** (utilizando repetições: for, while ou do...while) para inserir dados em cada variável de cada registro, via teclado, mostrando na tela a mensagem "Digite nr do aluno:", para que usuário entre com um numero, "Digite nome do aluno:", para que usuário entre com um nome, "Digite o departamento:", para que usuário entre com a abreviatura do departamento (CIC ou ENE), "Digite nota 0:", para que entrar com a nota 0, "Digite nota 1:", para que entrar com a nota 1 e "Digite nota 2:", para que entrar com a nota 2.

A sigla do departamento deve necessariamente ser armazenada em caixa alta (utilizar a função toupper). A primeira letra de cada nome deve ser maiúscula (toupper) e as demais minúsculas (tolower), mesmo que o usuário não respeite esta formatação. Ou seja, tanto no fornecimento da sigla do departamento como no fornecimento do nome, o usuário pode entrar com as informações sem respeitar qualquer critério (maiúscula ou minúscula), cabendo ao programa corrigir essa entrada, de acordo com o critério exposto anteriormente.

Percorrer cada registro do vetor de registros e calcule a média do aluno, a partir dos valores fornecidos para as notas 0, nota 1 e nota 2. Utilizar repetições (for, while ou do...while). O aluno não deve fazer $\text{media} = (\text{nota}[0] + \text{nota}[1] + \text{nota}[2]) / 3$.

1. Strings & Structs

Imprimir na tela os registros criados:

```
aluno[0].nr : 150  
aluno[0].nome : Ana  
aluno[0].curso : ENE  
aluno[0].nota[0]: 10.0  
aluno[0].nota[1]: 8.0  
aluno[0].nota[2]: 9.0  
aluno[0].media : 9.0
```

```
aluno[1].nr : 151  
aluno[1].nome : Beto  
aluno[1].curso : CIC  
aluno[1].nota[0]: 8.0  
aluno[1].nota[1]: 10.0  
aluno[1].nota[2]: 9.0  
aluno[1].media : 9.0
```

```
aluno[2].nr : 152  
aluno[2].nome : Carla  
aluno[2].curso : ENE  
aluno[2].nota[0]: 7.0  
aluno[2].nota[1]: 10.0  
aluno[2].nota[2]: 10.0  
aluno[2].media : 9.0
```

1. Strings & Structs

Ao final, criar mecanismos para responder o seguinte:

- A média da turma.
 - A menor nota na primeira prova.
 - O nome do aluno com maior média.
 - O numero de alunos abaixo da media da turma.
 - O(s) aluno(s) reprovado(s) foi(ram) : imprimir o(s) nome(s) do(s) aluno(s). Considere: reprovação \rightarrow média < 5.0). Caso não haja reprovação, imprimir "nao ha".
-

1. Strings & Structs

3. Implemente as soluções propostas para os exercícios 1 e 2 utilizando a linguagem Python [procure em Python uma alternativa para as *structs* em C].