

Prática sobre array de caracteres: Strings

```
// Exemplo de como usar o getchar

char nome[50]; //guarda o nome
char c;
int i, pos; // pos guarda a posição atual no vetor

//limpa o buffer antes de efetuar a leitura de valores
setbuf(stdin, NULL);

pos = 0;
c = getchar(); // busca uma letra no buffer
while (c != '\n') { // fica no loop enquanto não encontrar um ENTER
    nome[pos] = c;
    pos++;
    c = getchar();
}

for (i=0; i < pos; i++){
    printf("%c", nome[i]);
}
```

- 1) TABELA ASCII: o tipo char pode receber valores numéricos. Como ele ocupa 1 byte, os valores variam de -128 .. +127. Já o tipo unsigned char, que também ocupa 1 byte, representa números de 0 .. 255. Cada valor de um tipo char tem um caractere associado na tabela ASCII.

No comando printf, se você utilizar **%d** para mostrar uma variável **unsigned char**, será mostrado um número. Caso queira mostrar o caractere ASCII correspondente ao número armazenado na variável, utilize **%c**.

Monte um programa que mostra os valores e os caracteres correspondentes da tabela ASCII, para valores de 28 à 255

- 2) Utilizando os conceitos aprendidos até o momento (vetores e tipo char), crie um programa que leia o nome de uma pessoa e, em seguida, mostre o nome lido. Utilizar getchar() para entrada dos dados (veja acima um exemplo de como usar o getchar()). O nome pode ter no máximo 50 caracteres. Não usar as função de string.h

Exemplo de saída:

<< Vetor de Char>>

Digite um nome: **Bruno**

O nome digitado é: Bruno

- 3) Faça um programa semelhante ao anterior, só que o nome digitado deve ser mostrado em letras maiúsculas (pense como fazer isso olhando a diferença entre os valores dos caracteres minúsculos e maiúsculos na tabela ASCII). Não usar as função de string.h

Exemplo de saída:

```
<< Vetor de Char>>
```

Digite um nome: **Bruno**

O nome digitado é: BRUNO

- 4) Faça um programa em que o usuário digita dois nomes, e o programa informa se os nomes são iguais. Não usar as função de string.h

Exemplo de saída:

```
<< Compara nomes>>
```

Digite o nome 1: **Bruno**

Digite o nome 1: **Bruno**

Os nomes digitados são iguais

Exemplo de saída:

```
<< Compara nomes>>
```

Digite o nome 1: **Bruno**

Digite o nome 1: **BrunO**

Os nomes digitados são diferentes

- 5) Refaça o exercício anterior, para que agora não haja diferenciação entre caracteres maiúsculos e minúsculos.

Exemplo de saída:

```
<< Compara nomes>>
```

Digite o nome 1: **Bruno**

Digite o nome 1: **BrunO**

Os nomes digitados são iguais.

- 6) Faça um programa que recebe o nome e sobrenome de uma pessoa (máximo 40 caracteres), copie o nome para um vetor e o sobrenome para outro. Mostre os dois vetores copiados.

<< Vetor de Char>>

Digite um nome e sobrenome: **Bruno Travençolo**

Nome: Bruno

Sobrenome: Travençolo

- 7) O especificador de tipo %s pode ser usado para mostrar strings (vetores de caracter). Refaça o exercício 2 trocando o %c no printf pelo %s. Note que o loop sobre o vetor não é mais necessário. Descreva se o programa funciona com o %s.