

Arquitetura de Computadores 2010/11

Aula prática 3 – Tipos de dados em C

1. Considere as seguintes declarações:

```
int a = 2, b = 0;
```

Indique o resultado das seguintes expressões em C:

- a) `a == b`
- b) `!a || b`
- c) `!!a`
- d) `a = b`

Valide as suas respostas experimentalmente com um pequeno programa em C, que apresente no ecrã o resultado das diversas expressões.

2. Considere o seguinte programa:

```
#include <stdio.h>
int main()
{
    int i1 = -1;
    int i2 = -2;
    int i3 = -3;

    int *pi1 = &i1;
    int *pi2 = &i2;
    int *pi3 = &i3;

    *pi1 = -4;
    *pi2 = *pi1;

    pi2 = pi1;
    pi3 = pi2;

    *pi3 = -5;

    printf("%d\n%d\n%d\n", i1, i2, i3);

    return 0;
}
```

Diga quais os resultados que este programa imprime no ecrã. Confirme a sua resposta correndo o programa. Se a sua resposta não estiver certa tente perceber porquê.

Arquitetura de Computadores 2010/11

Aula prática 3 – Tipos de dados em C

3. Implemente uma função que apresente no ecrã os códigos ASCII de todos os caracteres de uma *string* dada como parâmetro.

Uma vez implementada a função, faça o programa principal que lê a *string* e apresenta os códigos ASCII, utilizando a função previamente desenvolvida.

Compile e teste o seu programa.

4. Implemente uma função que inverte uma *string*.

Dada a *string* (vector de caracteres), que a função recebe como parâmetro, esta deve proceder à inversão da posição de todos os caracteres até ao terminador '`\0`'. Note que o vector original será alterado automaticamente.

Uma vez implementada a função, faça o programa que lê uma *string* e a apresenta invertida no ecrã.

Compile e teste o seu programa

Analise os resultados obtidos.

5. Faça um programa que lê um número em base **B** (uma *string*) e apresenta no ecrã, o seu valor em decimal. Desenvolva o seu programa de forma incremental, seguindo os passos, aqui enumerados:

1. considera **B** = 2 (binário);
2. considera **B** ≤ 10, valor introduzido pelo utilizador;
3. considera **B** ≤ 16, valor introduzido pelo utilizador;
4. complete o seu programa verificando se a *string* introduzida é realmente um número na base **B**.

Não se esqueça que deve estruturar sempre o seu programa utilizando funções.

6. Faça o programa para decifrar a cifra de César com N=3. Decifre a seguinte frase – **Vdprv surjudpdu hp C!**