

## **Lista 2 Relacionada a Engenharia de Software I**

**Myllene Ferreira da Silva<sup>1</sup>**

<sup>1</sup>Departamento de Computação – DECOM –  
Universidade Federal Ouro Preto (UFOP) – Ouro Preto, MG – Brasil

[{myllene.silva@aluno.ufop.edu.br}](mailto:myllene.silva@aluno.ufop.edu.br)

### **1. Explique o funcionamento do mecanismo chamado “call-back function”.**

A função CallBack processa a mensagem lida pela aplicação, encerrando-a caso a aplicação receba uma mensagem WM\_DESTROY e caso receba uma mensagem que a aplicação não consegue processar, chama o procedimento padrão do Windows.

### **2. Explique o funcionamento do código contido no arquivo “main.cpp”.**

Código implementado para fazer manipulações nas janelas do Windows utilizando a biblioteca windows.

### **3. O que o código contido no arquivo “main1.cpp” faz?**

Ele transforma qualquer caractere minúsculo em maiúsculo, sendo ‘s’ o método de sair da execução.

### **4. Como este código funciona?**

O código ele utiliza a tabela ASCII para transformação de minúscula para maiúscula, sendo, subtraindo 32 sobre o valor referido ao caractere pela tabela ascii e assim retornando para o usuário.

### **5. Por que é necessário utilizar as instruções de “type casting”?**

Para converter uma variável para um tipo específico, sendo uma operação feita com o objetivo de alterar o tipo de um determinado valor.

### **6. O que significa “0xFF” no código “main2.cpp”?**

É um valor em formato de hexadecimal.

**7. O que o primeiro loop do código “main2.cpp” faz e como ele funciona?**

Ele utiliza um ponteiro que refere ao um vetor de char e faz um while para imprimir o conteúdo para o usuário.

**8. O que o segundo loop do código “main2.cpp” faz e como ele funciona?**

Ele imprime os valores do conteúdo do vetor de algarismo e endereço no qual o ponteiro aponta para cada posição do vetor.

**9. O que o terceiro loop do código “main2.cpp” faz? Por que ele não funciona?**

O terceiro loop do programa, imprime um vetor de caracteres que foi obtido através do “type casting” de um vetor de inteiros para caracteres, porém, esse loop não funciona de maneira adequada. Isso ocorre pois temos uma incompatibilidade entre o tipo do ponteiro e do conteúdo do vetor.

**10. O que pode ser concluído sobre a aritmética de ponteiros?**

Concluimos que os ponteiros podem ser usados para diversos tipos de variáveis mas se for inicializado de forma incorreta, o programa não será executado de maneira correta e podem ocorrer erros durante a execução.

**11. O código possui duas diretivas de pré-compilação que fazem com que o código “main3.cpp” gere erros. Compile e execute o código com a diretiva ERRO1 ativada e desativada e, então, baseado na saída impressa na tela, explique o que o trecho de código de linha 27 a 32 faz.**

O trecho de código recebe um ponteiro que é apontado para um endereço pré definido e após imprimir para o usuário, ele é modificado para apontar para o endereço 66.

**12. O código possui duas diretivas de pré-compilação que fazem com que o código “main3.cpp” gere erros. Compile e execute o código com a diretiva ERRO2 ativada e desativada e, então, baseado na saída impressa na tela, explique o funcionamento do trecho de código entre as linhas 37 e 57.**

O código faz uma verificação se pInt foi inicializado com NULL ou não, caso não seja inicializado, ele guarda um endereço de um lixo.