

Aula Prática 07 - Roteiro

21/05/2015 - Roteiro referente a aula prática 07 - Manipulação de Strings.

Todos os arquivos solicitados a seguir deverão ser criados no diretório "Aulas-Praticas" (ver aula 1).

1. Crie o arquivo "aula0701.h" contendo a definição do tipo *tipoErros* (tipo enumerado) e o protótipo da função *ObterMensagemErro*. Esta função deverá receber o número do erro e deverá retornar a mensagem (em português) correspondente ao erro. A macro referente à combinação *ifndef* e *define*, como por exemplo *_AULA0701_*, deverá ser definida como uma *string* valendo:

```
"@(#)aula0701.h $Revision$"
```

```
char *
ObterMensagemErro (tipoErros);
```

2. Crie o arquivo "aula0701.c" contendo a implementação da função *ObterMensagemErro*. Para a implementação desta função crie uma variável local estática (*static*) - *mensagensErro* - do tipo vetor de *strings* inicializada com as mensagens de erro necessárias.

Exemplo:

```
char *mensagensErro [ ] =
{
    "Operacao realizada com sucesso", /* mensagem correspondente ao codigo de retorno ok
(o) */
    "Argumento invalido" /* mensagem correspondente ao primeiro erro */
};
```

3. Crie o arquivo "aula0702.h" contendo o protótipo da função *GerarStringAleatoria*. Esta função deverá receber o conjunto de caracteres válidos e o comprimento desejado para a geração da *string*. Além disso, esta função deverá devolver a *string* gerada e deverá retornar ok (o) ou o código de erro correspondente. Este arquivo deverá conter também as macros e os tipos necessários para a implementação desta função. A macro referente à combinação *ifndef* e *define*, como por exemplo *_AULA0702_*, deverá ser definida como uma *string* valendo:

```
"@(#)aula0702.h $Revision$"
```

```
tipoErros
GerarStringAleatoria (char *caracteresValidos, unsigned comprimento, char
*stringGerada);
```

4. Crie o arquivo "aula0702.c" contendo a implementação da função *GerarStringAleatoria*.
5. Crie o arquivo "aula0703.c" contendo a implementação de um programa de testes para a função *GerarStringAleatoria*. Este programa deverá receber, através de dois argumentos da linha de comando, o conjunto de caracteres válidos e o comprimento desejado para a *string*. O programa deverá exibir a *string* gerada ou a mensagem de erro correspondente. Todos os tratamentos de erro necessários e que não possam ser realizados na função *GerarStringAleatoria* deverão ser implementados neste programa. Considere que o comprimento máximo solicitado para a *string* a ser gerada é igual a 1024 caracteres.
6. Inclua, no arquivo de dependências, a macro *AULA0703OBJ* - correspondendo aos arquivos necessários para criar o executável a partir dos arquivos "aula0701.c", "aula0702.c" e "aula0703.c". Além disso, defina a macro *AULA07*, equivalendo ao executável "aula0703", e altere a macro *EXECS*, de forma que o valor da mesma inclua os executáveis criados na aula 07. O arquivo de dependências deverá incluir ainda os objetivos *aula07* (todos os executáveis da aula 07) e *aula0703* (executável criado a partir dos arquivos definidos pela macro

AULA0703OBS) com os comandos correspondentes.

7. Compile, linkedite e teste o programa na máquina local (*CentOS*) e na *loghost02 (FreeBSD)*.
8. Submeta os arquivos "*aulao701.h*", "*aulao701.c*", "*aulao702.h*", "*aulao702.c*", "*aulao703.c*" e "*Makefile*" ao sistema de controle de versão *RCS* (comando *ci*).
9. Recupere uma cópia de leitura dos arquivos "*aulao701.h*" e "*aulao703.c*" (comando *co*) e uma cópia de escrita dos arquivos "*aulao701.c*", "*aulao702.h*", "*aulao702.c*" e "*Makefile*" (comando *co -l*).
10. Inclua, no arquivo "*aulao702.h*" o protótipo da função *ValidarCampoTipoString*. Esta função deverá receber uma *string*, o conjunto de caracteres válidos e os comprimentos mínimo e máximo permitidos para o campo. Além disso, esta função deverá retornar ok (o) ou o código de erro correspondente.

tipoErros

*ValidarCampoTipoString (char *string, char *caracteresValidos, unsigned minimo, unsigned maximo);*

11. Inclua, no arquivo "*aulao702.c*", a implementação da função *ValidarCampoTipoString*.
12. Crie o arquivo "*aulao704.c*" contendo a implementação de um programa de testes para a função *ValidarCampoTipoString*. Este programa deverá receber, através de quatro argumentos da linha de comando, uma string correspondendo ao valor do campo, o conjunto de caracteres válidos para o campo, além dos comprimentos mínimo e máximo para o campo. O programa deverá exibir se o campo é válido ou a mensagem de erro correspondente. Todos os tratamentos de erro necessários e que não possam realizados na função *ValidarCampoTipoString* deverão ser implementados neste programa
13. Inclua, no arquivo de dependências, a macro AULA0704OBS - correspondendo aos arquivos necessários para criar o executável a partir dos arquivos "*aulao701.c*", "*aulao702.c*" e "*aulao704.c*". Além disso, altere o valor da macro AULA07 de forma que inclua o executável "*aulao704*". O arquivo de dependências deverá incluir ainda o objetivo *aulao704* (executável criado a partir dos arquivos definidos pela macro AULA0704OBS) com os comandos correspondentes.
14. Submeta os arquivos "*aulao701.c*", "*aulao702.h*", "*aulao702.c*", "*aulao704.c*" e "*Makefile*" ao sistema de controle de versão *RCS* (comando *ci*).
15. Recupere uma cópia de leitura do arquivo "*aulao703.c*" (comando *co*) e uma cópia de escrita dos arquivos "*aulao701.c*", "*aulao702.h*", "*aulao702.c*" e "*Makefile*" (comando *co -l*).
16. Inclua, no arquivo "*aulao702.h*" o protótipo da função *GerarSenhaTemporaria*. Esta função deverá receber o comprimento desejado para a senha. Além disso, esta função deverá devolver a senha gerada e deverá retornar ok (o) ou o código de erro correspondente.

tipoErros

*GerarSenhaTemporaria (unsigned comprimento, char *senha);*

17. Inclua, no arquivo "*aulao702.c*", a implementação da função *GerarSenhaTemporaria*. A senha gerada deverá possuir sempre pelo menos uma letra maiúscula, uma letra minúscula, um dígito e um caractere especial (pertencente ao conjunto "*!@#\$%&_|*/-+=0[]<>,;:.*". Além disso, o comprimento mínimo para a senha é 8 caracteres.
18. Crie o arquivo "*aulao705.c*" contendo a implementação de um programa de testes para a função *GerarSenhaTemporaria*. Este programa deverá receber, através de um argumento da linha de comando, o comprimento desejado para a senha. O programa deverá exibir a senha gerada ou a mensagem de erro correspondente. Todos os tratamentos de erro necessários e que não possam realizados na função *GerarSenhaTemporaria* deverão ser implementados neste programa
19. Inclua, no arquivo de dependências, a macro AULA0705OBS - correspondendo aos arquivos necessários para criar o executável a partir dos arquivos "*aulao701.c*", "*aulao702.c*" e "*aulao705.c*". Além disso, altere o valor da macro AULA07 de forma que inclua o executável "*aulao705*". O arquivo de dependências deverá incluir ainda o objetivo *aulao705* (executável criado a partir dos arquivos definidos pela macro AULA0704OBS) com os comandos correspondentes.
20. Submeta os arquivos "*aulao701.c*", "*aulao702.h*", "*aulao702.c*", "*aulao705.c*" e "*Makefile*" ao sistema de controle de versão *RCS* (comando *ci*).
21. Recupere uma cópia de leitura dos arquivos "*aulao701.c*", "*aulao702.h*", "*aulao702.c*" e "*aulao705.c*" (comando *co*) e uma cópia de escrita do arquivo "*Makefile*" (comando *co -l*).

Observações:

- Todos os arquivos criados deverão incluir comentários com as seguintes informações:
Universidade Federal do Rio de Janeiro
Escola Politecnica
Departamento de Eletronica e de Computacao
EEL270 - Computacao II - Turma 2015/1
Prof. Marcelo Luiz Drumond Lanza
Autor: Nome Completo

\$Author\$
\$Date\$
\$Log\$
- No final do arquivo inclua um comentário com a macro \$RCSfile\$.
- Prazo: 28/05/2015 (início da aula prática 08).