

MC102: Primeira aula de laboratório

2016

Instruções iniciais

- Laboratórios
 - Em geral de um a dois laboratório por semana cobrindo tópicos vistos em aulas anteriores.
 - Laboratórios entram no ar sempre nas segundas-feiras, e deverão ser entregues até a sexta-feira da semana seguinte (12 dias de prazo).
 - **Os prazos de entrega não serão estendidos, mesmo que hajam feriados na semana de entrega!**

Susy

- Sistema de submissão e testes automáticos de programas.
- Desenvolvido pelo Prof. Tomasz Kowaltowski (Instituto de Computação - Unicamp).
- **Faz detecção de plágio! Fraudes implicam em nota 0 no semestre!**
- Nesta primeira aula veremos um exemplo de submissão.

- Programas importantes:
 - Terminal: interface para execução de comandos do sistema operacional.
 - gedit: editor de texto.
 - codeblocks: sistema integrado para desenvolvimento de programas.
 - firefox: navegador web.
 - gcc: compilador C.

Alguns comandos no Linux [1/2]

- `cd`: muda para um diretório.
 - `$ cd Documents`
Muda para o diretório Documents.
- `mkdir`: cria um novo diretório.
 - `$ mkdir Lab00`
Cria o diretório Lab00.
- `pwd`: identifica o diretório atual.
 - `$ pwd`
`/home/ra012345/Lab00`

Alguns comandos no Linux [2/2]

- ls: lista o conteúdo de um diretório.
 - \$ ls Documents
aprendendoC.pdf algoritmos.pdf lista1.pdf
Lista o conteúdo do diretório Documents.
- mv: move arquivos para um diretório.
 - \$ mv Downloads/maq01.in Lab00
Move o arquivo maq01.in que está no diretório Downloads para o diretório Lab00.
- diff: compara dois arquivos.
 - \$ diff maq01.out maq01.res
Compara os arquivos maq01.out e maq01.res

Escrevendo e Compilando um Programa pelo Terminal

Atividade:

- 1 Abra um terminal:
 - Aplicações → Acessórios → Terminal
 - Applications → Systems Tools → Terminal
- 2 Crie uma pasta para salvar seus programas.
 - Exemplo: LabsMC102.

Escrevendo e Compilando um Programa pelo Terminal

Atividade:

- 1 Abra o *codeblocks* ou um editor de texto como o *gedit*.
- 2 Copie o programa que será projetado no editor.
- 3 Salve o arquivo com extensão ".c" na pasta criada.

Escrevendo e Compilando um Programa pelo Terminal

- 1 No terminal, mude para a pasta onde foi salvo o programa.
 - `$ cd LabsMC102`
- 2 Utilize o comando `gcc` para compilar o programa.
 - `$ gcc -std=c99 -pedantic -Wall -o exemplo exemplo.c`
`$ ls LabsMC102`
`exemplo exemplo.c`
 - O arquivo "exemplo"gerado pela compilação acima é um executável, que pode ser executado pelo seguinte comando.
 - `$./exemplo`

Enviando um programa para o SuSy

- 1 Entre na página da disciplina.
- 2 Acesse a página de tarefas da disciplina no SuSy através do link especificado na página da disciplina.
- 3 Acesse a página da tarefa desejada.
- 4 Informe o seu nome de usuário e sua senha e escolha o(s) arquivo(s) a ser(em) enviado(s).
- 5 Clique em "Submeter".

Consultando uma submissão no SuSy

- 1 Acesse a página da tarefa desejada no SuSy.
- 2 Informe seu nome de usuário e sua senha.
- 3 Clique em “Consultar” - será a mostrada uma tela com os dados da sua última submissão.

Recuperando um programa submetido ao SuSy

- 1 Acesse a página da tarefa desejada no SuSy.
- 2 Informe seu nome de usuário e sua senha.
- 3 Clique em “Recuperar” - será mostrada uma tela com um link para o(s) seu(s) último(s) arquivo(s) submetido(s).

Testando um programa antes de submeter ao SuSy

- 1 Baixe os arquivos de entrada dos testes da tarefa (arquivos nomeados da forma `arqX.in`).
- 2 Baixe os arquivos de resposta dos testes da tarefa (arquivos nomeados da forma `arqX.res`).
- 3 Mova todos os arquivos baixados para o diretório onde está o executável do seu programa.
- 4 Para cada teste, execute o seu programa passando o arquivo de entrada e salvando a saída (em um arquivo nomeado da forma `arqX.out`)
 - `$./exemplo < arq01.in > arq01.out`
- 5 Para cada teste, compare a saída do seu programa com a resposta do teste
 - `$ diff arq01.out arq01.res`

Agradecimentos

- Laboratórios
 - Esta apresentação foi feita com uso do material preparado por Ivan Sichmann Freitas, Sérgio Durigan Júnior e Gabriel Krisman Bertazi para aulas introdutórias sobre Linux e Software Livre (https://github.com/ISF/aulas_gpsl)
 - Esta apresentação também foi baseada em uma apresentação feita Por Danilo Marshall