

Projeto

1. Objetivo

O objetivo deste projeto é que os alunos apliquem conceitos aprendidos no decorrer do curso na criação de um *shell script* que resolva algum problema do mundo real que seja interessante e útil.

2. Conceitos

A tabela a seguir detalha 20 (vinte) conceitos que poderão ser utilizados no *shell script* que será desenvolvido. Embora o *script* possa fazer uso de outros conceitos alheios aos listados abaixo, serão considerados na contabilização da nota somente aqueles presentes nessa tabela.

#	Conceito	Aula
01	Comandos: <code>echo, ls, pwd, cd, tree, find, locate, whereis, file, stat, date, cal, history, clear, logout</code> ou <code>exit</code>	Comandos básicos
02	Comandos: <code>strings, cat, tac, wc, more, less, head, tail, tee, sort, zcat</code> ou <code>diff</code>	Processamento de texto
03	Comandos: <code>mkdir, cd, rmdir, mv, cp, mv, ln, id, umask, basename, dirname, gzip, gunzip, tar, split, shred, cksum, md5sum, dd, lsof</code> ou <code>fuser</code>	Sistema de arquivos
04	Permissões em representação octal ou simbólica: <code>chmod</code>	
05	Comandos: <code>ps, top, jobs, bg, fg, pstree, vmstat, xload</code> ou <code>tload</code>	Processos
06	Sinais: <code>kill</code> ou <code>killall</code>	
07	Redirecionamentos da entrada, saída ou saída de erro padrão	Entradas, saídas e Redirecionamentos
08	<i>Pipes</i> ou redirecionamentos avançados	
09	Expansão de arquivos, til (~), aritmética, de chaves, de parâmetros, substituição de comandos e escapes	Expansões
10	Variáveis: criadas pelo usuário ou reservadas	
11	Parâmetros posicionais e especiais: <code>\$0</code> até <code>\$n</code> , <code>\$*</code> , <code>\$@</code> , <code>\$#</code> , <code>\$?</code> , <code>\$\$</code> , <code>\$!</code>	
12	Atributos de variáveis: arranjo, função, inteira, somente escrita (<code>readonly</code>) ou global (<code>export</code>)	
13	Expansões avançadas: <code>\${name:-value}</code> , <code>\${name:=value}</code> , <code>\${name:?value}</code> , <code>\${name:+value}</code> , <code>\${parameter:offset}</code> , <code>\${parameter:offset:length}</code> , <code>\${#parameter}</code> , <code>\${!parameter}</code> , <code>\${parameter#word}</code> , <code>\${parameter##word}</code> , <code>\${parameter%word}</code> , <code>\${parameter%%word}</code> , <code>\${parameter/pattern/string}</code> ou <code>\${parameter//pattern/string}</code>	Parâmetros e Variáveis
14	Testes condicionais com <code>test</code> ou <code>if</code> : expressões com arquivos, strings, inteiros ou expressões regulares (<code>=~</code>)	Estruturas de fluxo condicionais
15	Testes condicionais com <code>case</code>	
16	Outras formas de desvio: <code>&&</code> ou <code> </code>	
17	Comandos ou sequenciadores: <code>for, while, until, select, break</code> ou <code>continue</code>	Estruturas de repetição

18	Expressão regular básica ou estendidas: <code>grep</code> ou <code>egrep</code> OBS: a expressão regular deve usar pelo menos um metacaractere	Expressões Regulares
19	Programa em <code>sed</code>	<code>sed</code>
20	Programa em <code>AWK</code>	<code>AWK</code>

3. Instruções

O projeto será dividido em três etapas de acordo com o seguinte calendário:

Etapa	Descrição	Datas
01	Definição do problema a ser resolvido no projeto.	02/12/2024
02	Relatório contendo a descrição do problema resolvido, código do <i>shell script</i> desenvolvido e planilha com os conceitos utilizados no formato definido.	06/01/2024
03	Apresentação do projeto para a turma.	09/01/2024 a 11/02/2025

Na primeira etapa, cada aluno deverá enviar para a lista de discussão (decom042@googlegroups.com) uma descrição do problema que se deseja resolver. O intuito é que se resolva algum problema do mundo real que seja interessante e útil. **O problema escolhido deve, obrigatoriamente, obter informações de alguma fonte externa (arquivo, internet, etc) e/ou produzir alguma informação externa (documento, relatório, etc).**

Na segunda etapa, cada aluno deverá **preparar um relatório em formato PDF** que será submetido pelo SIGAA (<https://sig.cefetmg.br/sigaa>). O relatório deverá conter: (1) a descrição do problema que foi definido na lista de discussão, (2) o código do *shell script* desenvolvido para resolver esse problema com suas linhas numeradas, e (3) uma planilha com 10 (dez) conceitos, dos 20 (vinte) expostos na seção 2. O *shell script* deverá ter no mínimo 30 (trinta) linhas de código, verificadas pela ferramenta `cloc` (<https://github.com/AlDanial/cloc>). Cada conceito utilizado na planilha deve ser único e estar acompanhado da aula que o envolve, o trecho de código com número da linha que o utiliza e uma breve explicação. **Os conceitos escolhidos devem ser úteis para a resolução do problema.** Por exemplo, se um *script* possui a atribuição `c=$((3*x+1))` na linha 5, a planilha poderia ser preenchida conforme a seguir. **Note que um mesmo trecho de código pode ser utilizado em mais de um conceito, mas cada conceito deve ser único, ou seja, se seu *script* utilizar expansão de chaves em outra parte do código, você não pode usar o conceito 09 novamente.**

Conceito	Aula	Linha	Trecho	Descrição
...
09	Expansões	5	<code>c=\$((3*x + 1))</code>	Expansão aritmética com <code>c=\$(())</code> .
10	Parâmetros e variáveis	5	<code>c=...</code>	Variável <code>c</code> definida pelo usuário.
...

Na terceira etapa, serão realizadas apresentações conforme calendário a seguir. **As apresentações seguirão a ordem em que os problemas foram enviados para a lista de discussão, ou seja, os primeiros a enviarem serão os primeiros a apresentarem.** Você deve explicar o problema resolvido, mostrar o *script* desenvolvido com sua execução se possível e os conceitos escolhidos com explicação. As apresentações devem ter no máximo 20 minutos.

Data	Aluno(a)
14/01/2025	MARINA BERNARDES DINIZ
	André Oliveira Martins
	SAMUEL FERREIRA LOPES
	JOAO PEDRO DE ALMEIDA CAMPOS
16/01/2025	NICOLAS RODRIGUES DE VARGAS
	ALEX MEIRELES SANTOS ALMEIDA
	LUIZ FELIPE RODRIGUES DOS SANTOS
	Rosane Silva Freitas Araujo
21/01/2025	JOÃO GUIMARÃES MOREIRA
	LUCAS DA SILVEIRA NASCIMENTO
	YURI BARBOSA CARVALHAIS
	LUCCA MOLINARI BARBOSA
23/01/2025	Pedro Henrique Ribeiro
	PEDRO HENRIQUE DE MORAES SOARES
	JOÃO MANOEL ROCHA NASCIMENTO
	ITALO GUSTAVO DONATO CORDEIRO
28/01/2025	DIOGO PEDROSA MIRANDA
	PAULO HENRIQUE SILVA

4. Avaliação

O projeto será avaliado em 20 (vinte) pontos, sendo 2 (dois) pontos para o problema enviado para a lista de discussão, 13 (treze) pontos para o relatório e 5 (cinco) pontos para a apresentação. A cada dia de atraso no envio do problema para a lista de discussão serão descontados 0,25 (zero, vinte e cinco) pontos. A presença nas apresentações é obrigatória, ou seja, caso o(a) aluno(a) falte será descontado 1 (um) ponto para cada dia de apresentação perdida.