



Universidade Federal de Uberlândia
Faculdade de Engenharia Elétrica

Tarefa de Sistemas Embarcados 2

Aluno: Angellus Uriãh de Freitas França

nº de matrícula: 11911EAU001

Professor: Éder Alves

20 de outubro de 2022

Conteúdo

1 Questão 1	3
2 Questão 2	3
3 Questão 3	4
4 Questão 4	4
5 Referências	5

1 Questão 1

A questão 1 consiste em assistir quatro vídeos, como é mostrado a seguir.

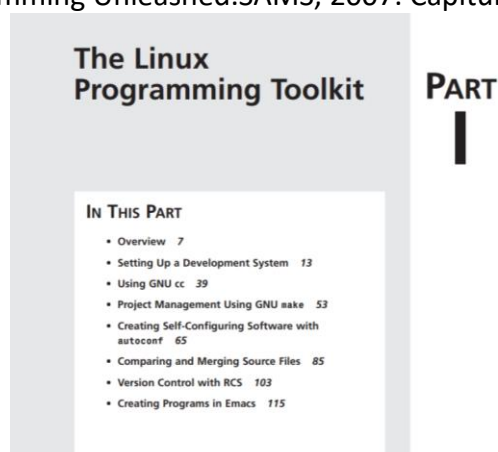
1. Veja os seguintes vídeos:

- <https://www.youtube.com/watch?v=rrB13utjYV4>
- <https://www.youtube.com/watch?v=l4EWvMFj37g>
- <https://www.youtube.com/watch?v=U3aXWizDbQ4>
- <https://www.youtube.com/watch?v=MNeX4EGtR5Y>

Os vídeos consistem em apresentar um pouco do Linux, Bash, C, C++ e suas histórias em 100 segundos.

2 Questão 2

A questão 2 consiste em utilizar o material de suporte do livro Kurt Wall. Linux Programming Unleashed.SAMS, 2007. Capítulo 3, e responder as seguintes perguntas:



Liste e descreva o que são as 4 etapas do processo de compilação.

R: • Preprocessing: Essa é a primeira etapa para um compilador, onde o código fonte é incluído, analisado por erros de sintaxe, e macros ou definições são substituídos e processados. Essa etapa ocorre muito em linguagens como C e C++. • Compilation Proper: Nessa etapa, o código fonte é transformado em um código assembly, um código muito próximo do de máquina, ou código binário. Contudo, ainda contém referências a arquivos externos, então não pode ser utilizado. • Assembly: Com o código assembly pronto, ele passa por um conversor, chamado assembler, para se tornar um código binário feito exclusivamente para um único sistema. • Linking: Essa é a última etapa do compilador, onde as bibliotecas, já compiladas, são adicionadas em nosso código binário, permitindo a criação de um arquivo binário executável.

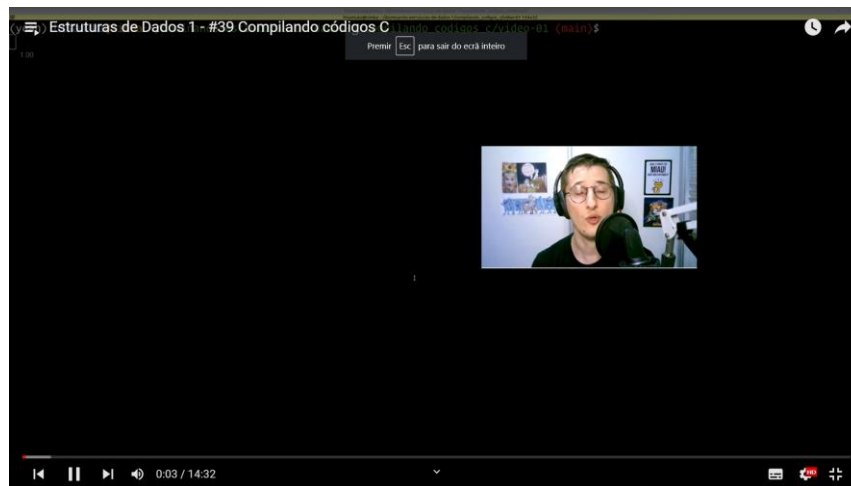
b)O compiladorgcc permite fornecer parâmetros extras, que modificamdesde a emissão de erros atéo binário final, o otimizando para determinados comportamentos. Explique a função dos seguintes parâmetros:

i) -static: Link para bibliotecas estáticas.

- ii) -g: Incluir informações de depuração padrão no binário
- iii) -pedantic: Emite todos os avisos exigidos pelo padrão ANSI/ISO C Standard
- iv) -Wall: Emite todos os avisos geralmente úteis que o gcc pode fornecer. Avisos específicos também podem ser sinalizados usando -W{warning}.
- v) -Os: A opção bare -O diz ao gcc para reduzir o tamanho do código e o tempo de execução
- vi) -O3: As opções -O3 incluem todas as otimizações de O2, desenrolamento de loop e outros recursos específicos do processador.

3 Questão 3

A questão 3 consiste em assistir os 5 primeiros vídeos da lista disponível no link:
<https://www.youtube.com/watch?v=hrPwxKtedCc&list=PL3Zsll15yo2pCf0WpZmV-ga02kMPxKH3p&index=1>

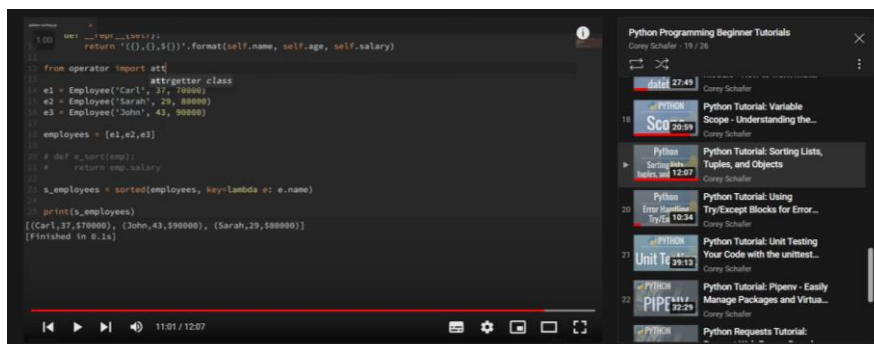


Além disso, deve ser reproduzido todos os exemplos feitos no vídeo e upar no Github.

4 Questão 4

Assim como na questão 3, é necessários assistir os vídeos da lista disponível no link a baixo, reproduzir os exemplos feitos e upar no Github.

<https://www.youtube.com/watch?v=k9TUPpGqYTo&list=PLosiE80TeTskrapNbzxhwoFUIlCjGgY7&index=2>



Referências

Kurt Wall. Linux Programming Unleashed.SAMS, 2007. Capítulo 3