

Uso de linha de comando, pastas, arquivos e alocação de memória em linguagem C: estudo de caso com um gerador índice remissivo

Cronograma

Entrega: 09/06/2022

Dúvidas: 19/05/2022 e 02/06/2022

Equipes e reservas de programas: 19/05/2022

Enunciado: 19/05/2022

Objetivo

Aplicar idiomas e práticas da programação em **linguagem C** no contexto de **software básico** na escrita de um utilitário que gera **índices remissivos**.

Atenção: não são aceitos programas escritos em linguagem C++, C#, Objective-C ou similares.

Enunciado

Um gerador de índice remissivo é uma ferramenta de indexação de textos impressos [3]. A principal característica desses programas é o processamento de um número indeterminado de palavras diferentes. Ou seja, somente durante a execução é possível determinar a área necessária para armazenar a estrutura de dados necessária. Em linguagem C, isso significa que a memória deve ser alocada de forma dinâmica (malloc) para permitir que a estrutura de dados do índice remissivo possa utilizar o espaço do "heap". Lembre dos demais processos, não reserve memória acima do necessário. Considere a realocação de um bloco de memória de acordo com a necessidade ou o uso de uma estrutura encadeada. A operação básica do programa *ig* (*index generator*) deve ser capaz de gerar uma lista de palavras, com indicação de página e/ou parágrafo onde ocorrem no texto, a partir de um arquivo de texto. A entrada e a saída podem indicar arquivo de entrada e arquivo de saída.

O programa deve oferecer ajuda ("usage") e indicar sua versão.

O programa deve oferecer ao menos dois formatos de saída. Um dos formatos de saída deve ser em texto (ex. CSV, JSON) e um outro deve ser um formato gráfico (HTML, SVG).

Exemplos de uso:

```
% ig --csv machado.txt --html dom.html dom_casmurro.txt
```

```
% ig --help
```

O programa deve adotar as convenções de organização do projeto GNU [1][2]. Uma compilação modular com Makefile deve ser adotada com ao menos um arquivo de rotinas, com arquivo de cabeçalho e arquivo principal.

O grupo deve propor um usuário específico (pesquisador, historiador, jornalista) e implementar funções que possam ser úteis em um contexto específico. Por exemplo, um historiador pode usar um índice remissivo para determinar se existem ocorrências de um nome ou palavra em um documento.

Funções adicionais incluem lista de exclusão, lista de inclusão, lista de expressões compostas, escolha de cores e fonte, frequência mínima e sinônimos [4]. A estratégia mais comum é fornecer uma lista de palavras que devem ser localizadas e compor o índice. O grupo deve

consultar e indicar programas similares que servem como referência do uso e da implementação desse tipo de software.

Exemplos de algumas **possibilidades** de uso desejado para o programa solicitado:

```
% cat dom_casmurro.txt > ig --csv
```

```
% ig --excludes nono.txt --case-sensitive --min 30 --svg -o dom.svg dom_casmurro.txt
```

O trabalho deve ser mantido em um repositório no GitHub. A formação de grupos e o repositório devem ser indicados no Moodle. É recomendado utilizar uma ferramenta que inspecione o código-fonte, a programação em linguagem C é desafiadora [6].

Entrega

Cada grupo deve realizar uma apresentação de até 20 minutos, com a apresentação dos itens desenvolvidos nos critérios de avaliação indicados a seguir.

O grupo deve realizar a gravação e entregar o endereço do vídeo hospedado em um sistema de compartilhamento de mídias (ex.: Zoom e YouTube).

Critérios de avaliação

Identificar e destacar na apresentação do trabalho:

- a) [1 pt] Ajuda e versão: indicar ajuda e versão quando solicitado, conforme convenções da GNU.
- b) [1 pt] Compilação modular: Makefile, arquivo de rotinas e arquivo de cabeçalhos.
- c) [2 pt] Alocação no "heap": uso de ponteiros e alocação fora do "stack".
- d) [1 pt] Ocorrência simples: saída em texto com palavras e ocorrências, arquivo de saída pode ser lido em um aplicativo de planilha eletrônica.
- e) [2 pt] Visualização: saída gráfica com destaque para cores e tamanhos de fonte, arquivo de saída deve ser lido em um navegador Web.
- f) [1 pt] Repositório: repositório com ao menos uma semana de atividade e *commits* de todos os integrantes.
- g) [1 pt] Exemplo de uso: explicar o motivo do usuário necessitar de um índice remissivo.
- h) [1 pt] Função adicional: parâmetro de entrada com função adicional proposta pelo grupo, veja exemplos no enunciado.
- i) [1 pt] Inspeção automática de código: utilizar GitHub Actions para avaliar a qualidade do código com "linter" ou similar [5].

A avaliação do trabalho é limitada em um total de 10 pontos.

Referências

- [1] GNU. Core Utils <https://github.com/coreutils/coreutils/tree/master/src>
- [2] GNU. Savannah <https://git.savannah.gnu.org/cgit/>
- [3] WIKIPÉDIA. Índice remissivo. https://pt.wikipedia.org/wiki/%C3%8Dndice_remissivo
- [4] WIKIPÉDIA. Index [https://en.wikipedia.org/wiki/Index_\(publishing\)](https://en.wikipedia.org/wiki/Index_(publishing))
- [5] Super Linter. <https://github.com/github/super-linter>