

Projeto – Mini Googlebot (Parte II)

Data de entrega: 07/12. Será descontado 1 ponto por dia de atraso.

Grupos de 2 alunos (pode ser diferente do grupo do Projeto 1)

1. Objetivo

O objetivo deste projeto é desenvolver um Mini Googlebot, isto é, uma simulação do algoritmo Google real para captação de informações. O Google utiliza um robô, chamado Googlebot, para varrer a web em busca de links que apontem para sites. O resultado é um conjunto de informações de cada site, permitindo que o algoritmo de busca do Google faça computações sobre as mesmas decidindo a relevância (ou não) de cada site para a busca do usuário.

O objetivo específico desta etapa do projeto é desenvolver a operação de busca, conforme especificado no item 2, integrando a solução à etapa anterior do projeto.

2. Especificação

Considerando a Parte I do projeto, inclua as novas funcionalidades especificadas abaixo.

- **Busca por palavra-chave:** dada uma palavra, o sistema deve pesquisá-la entre as palavras-chave de cada site e retornar o nome e link dos sites que a contém. O resultado deve ser ordenado por relevância de modo decrescente.

Esta é a operação mais importante e frequente no sistema Mini Googlebot. Novamente cabe ao grupo definir a melhor estrutura de dados a ser empregada para a busca e também como integrar a nova funcionalidade à implementação desenvolvida na parte I do projeto.

- **Sugestão de sites:** sites relacionados à busca realizada. Essa operação inclui 4 passos: a) buscar a palavra fornecida entre as palavras-chave de cada site, selecionando aqueles que a contém; b) coletar todas as palavras-chave dos sites selecionados no passo “a”; c) buscar os sites que contêm alguma das palavras-chave identificadas no passo “b”; d)

mostrar o nome e o link dos cinco (5) sites mais relevantes ordenados por relevância (decrecente).

3. Ferramentas, modelo de desenvolvimento e de entrega

O projeto deve ser desenvolvido em grupos de 2 alunos. A implementação deve ser realizada em linguagem C, padrão C99.

Deve ser submetido um arquivo compactado (.zip) RunCodes de um dos alunos do grupo. O arquivo .zip deve conter: os fontes (arquivos .c e .h), um Makefile que compile o projeto, um arquivo README. O projeto pode estar organizados em pastas/diretórios, desde que o Makefile seja configurado para fazer a compilação adequadamente. O arquivo README deve conter: os nomes de todos os integrantes do grupo; compilador/plataforma utilizado; instruções de uso do programa e justificativas para as escolhas das estruturas assim como justificativas para as implementações escolhidas para as estruturas e de seus métodos (inserção, remoção e busca).

4. Critérios de Avaliação

- Aderência à especificação e ao uso de TADs (5 pontos).
- Eficácia e eficiência da solução (4 pontos).
- Documentação (1 ponto).
- Funcionalidades (facilidades de uso do programa – 1 ponto) (Bônus – Importante: o bônus só é válido para os grupos com nota máxima em aderência à especificação).

Observações:

- 1- Os projetos serão compilados e corrigidos utilizando o compilador GCC Linux.
- 2- Haverá, obviamente, verificação de plágio. Aos projetos envolvidos em plágio será atribuídas nota zero.

Bom Trabalho!