

**Universidade Federal do Espírito Santo – Departamento de Informática**  
**Estruturas de Dados I (INF09292)**

**1o Trabalho Prático**

**Período: 2020/1 EARTE**

**Profª Patrícia Dockhorn Costa**

**Email: [pdcosta@inf.ufes.br](mailto:pdcosta@inf.ufes.br)**

***Data de Entrega: 23/11/2020 - Trabalho individual***

***Este trabalho tem como objetivo praticar o uso de tipos abstratos de dados e estruturas do tipo Lista.***

***Regras Importantes***

- Não é tolerado plágio. Trabalhos copiados serão penalizados com zero.
- A data de entrega é inadiável. Para cada dia de atraso, é retirado um ponto da nota do trabalho.

***Material a entregar***

- Documentação do trabalho, que deve conter:
  - Capa do Projeto: deve conter os seguintes itens: Título, Autoria e Data.
  - Introdução: descrição do problema a ser resolvido e visão geral sobre o funcionamento do programa (em termos de módulos - TADs, arquivos, etc.).
  - Metodologia: descrição da implementação do programa. Devem ser detalhadas as estruturas de dados utilizadas (preferencialmente **com diagramas ilustrativos**), o funcionamento das principais funções utilizadas, bem como decisões tomadas relativas aos casos e detalhes de especificação que porventura estejam omissos no enunciado. **Modularize o seu programa usando a técnica de tipos abstratos de dados**, como discutido em sala de aula.
  - Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação.
  - Referências bibliográficas: bibliografia utilizada para o desenvolvimento do trabalho, incluindo sites da Internet se for o caso.
- Pelo Classroom:
  - Documentação do trabalho (em formato PDF).
  - Todos os arquivos .c e .h criados (espero código muito bem documentado!). Favor não enviar arquivos de projetos, apenas os códigos-fonte!
  - O makefile (ou instruções de como compilar/linkar seu programa na linha de comando).

## WikED!

Nas últimas décadas, um grande número de sistemas colaborativos apareceram na Web. Exemplos clássicos são a *Wikipedia*, *Linux* e *Yahoo! Answers*. Nestes sistemas, usuários finais colaboram para construir algum artefato que será útil para toda a comunidade. No caso da *Wikipedia*, em particular, seus usuários finais são responsáveis por editar colaborativamente as páginas da enciclopédia. Muitas pessoas colaboram diariamente e milhares de páginas são editadas por hora. Todas as mudanças são armazenadas em históricos, o que permite a visualização dos editores e das modificações feitas nas páginas.

Neste trabalho de estruturas de dados, nós vamos implementar um sistema que simula a construção colaborativa de uma enciclopédia similar à *Wikipedia*, chamada de *WikED!*.

Chamaremos de “Página” os artigos que serão editados colaborativamente pelos usuários, que exercem, neste contexto, o papel de “Editores”. As páginas serão construídas a partir de “Colaborações”, ou seja, pedaços de texto fornecidos pelos editores. Portanto, editores poderão contribuir com colaborações para diferentes páginas. Além de colaborações, páginas também podem possuir “Links” para outras página. Será importante manter o histórico de colaborações e links, para que tenhamos um panorama, a qualquer instante, dos editores e das modificações das páginas.

Podemos implementar a WikED! em C utilizando um conjunto de listas encadeadas. A Figura 1 ilustra uma sugestão de organização de listas encadeadas para implementar a WikED! (a figura sugere uma organização, mas não a impõe).

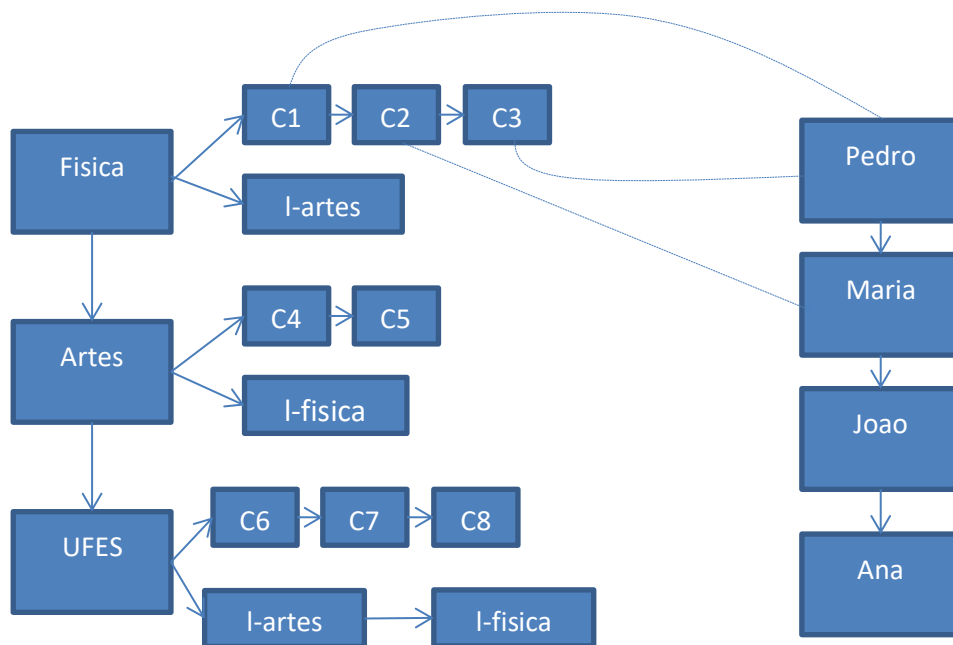


Figura 1 - Possível Implementação da WikED!

Na WikED! apresentada na Figura 1, há apenas 3 páginas cadastradas: Física, Artes e UFES. Há 4 editores: Maria, Joao, Pedro e Ana. A página de física recebeu 3 contribuições (C1, C2 e C3). C1 e C3 foram fornecidas pelo editor Pedro, enquanto que C2 pela editora Maria. Contribuições são trechos de texto que serão fornecidos pelos editores por meio de arquivos. Por questões de simplicidade, a figura não mostra as outras contribuições dos editores. Mas considere que C4 foi fornecida por Maria; C5 e C8 por Ana; e, C6 e C7 por Joao.

Veja que, além de uma lista de contribuições, as páginas também possuem uma lista de “links”, que configuram os links entre as páginas. Por exemplo, a página da UFES possui links para as páginas de Física e Artes, enquanto que a página de Física possui link para a página de Artes e a de Artes para a de Física.

Neste trabalho, você deverá implementar a WikED!. Faz parte do trabalho projetar os Tipos Abstratos de Dados necessários (usando tipos opacos), bem como implementá-los. Como padrão, insira os elementos SEMPRE ao final das listas.

### **Considerações Importantes**

- Nomes de Páginas e Editores serão simples e sem acentuação, nunca nomes compostos.
- Páginas e Editores deverão ter nomes que os identifiquem unicamente.
- Páginas podem ser excluídas. Quando for o caso, deve-se excluir todas as contribuições relativas a ela, bem como os links para outras páginas. Além disto, deve-se percorrer a WikED! para excluir possíveis links de outras páginas para esta.
- Editores também podem ser excluídos. Neste caso, deve-se percorrer a WikED! para excluir todas as contribuições deste editor (estas não serão mantidas no histórico da WikED!).
- Contribuições também podem ser “retiradas”. Neste caso, deve-se manter o registro da retirada, para o histórico de contribuições. Ou seja, não será excluída a contribuição da lista de contribuições (algum campo informativo deve dizer se a contribuição foi retirada, ou não). Lembrando que apenas o Editor que inseriu a contribuição terá o direito de excluí-la.
- Deve ser possível inserir e retirar links de páginas.

### **Impressão de uma Página WikED!**

Quando for pedido para imprimir uma determinada página (será fornecido o nome do arquivo), deve-se abrir o arquivo e incluir as seguintes informações:

Página <nome da pagina>

Histórico de Contribuições: Lista de contribuições (nomes dos arquivos), com os nomes dos respectivos editores responsáveis. Caso a contribuição tenha sido retirada pelo editor, deve-se informar que a contribuição foi retirada por seu respectivo editor.

Lista de Links: links para outras páginas

Textos: Trechos de texto fornecidos pelos editores, na ordem com que foram incluídos (as contribuições retiradas não devem ser consideradas).

### Arquivo de Log

Como saídas do trabalho, além dos arquivos com as páginas da WikED!, será pedido um arquivo de “log” para possíveis mensagens de erro e para informar se há caminho entre duas páginas (por meio de links). No caso de mensagens de erro, por exemplo, se um editor tentar excluir uma contribuição que não tenha sido inserida por ele/ela, uma mensagem de erro deve ser gerada (e.g.: erro: editor não tem direito de excluir esta contribuição). Verificações similares devem ser feitas caso seja pedido para excluir uma página inexistente, ou um editor inexistente, etc. No caso de saber se há caminho entre duas páginas, basta navegar nas listas de links para determinar se há caminho, ou não entre as páginas. No exemplo da figura 1, há caminhos da página da UFES para as páginas de Física e Artes, mas não há caminho inverso.

### Execução do programa

O programa testador deverá ser capaz de ler as instruções do arquivo texto de entrada e realizar as devidas operações na WikED!. O seu programa testador deverá ler os dados de entrada a partir de um arquivo, cujo nome é passado como parâmetro por meio de **argc** e **argv** (faz parte do trabalho descobrir como manipular arquivos e strings em C). Exemplo de execução do programa a partir da linha de comando: *./wiked.out entrada.txt*

O arquivo de entrada é basicamente uma lista de comandos (um por linha) em formato texto. O formato a ser usado é exemplificado abaixo (considere que todos os arquivos textos, tanto de entrada, quanto de saída, estão no mesmo diretório do executável):

### **Exemplo de arquivo de entrada**

```
INSEREPAGINA Fisica fisica.txt
INSEREPAGINA Artes artes.txt
INSEREPAGINA UFES ufes.txt
RETIRAPAGINA Psicologia
INSEREEDITOR Pedro
INSEREEDITOR Maria
INSEREEDITOR Joao
INSEREEDITOR Ana
INSEREEDITOR Edu
RETIRAEDITOR Edu
INSERECONTRIBUICAO Fisica Pedro c1.txt
INSERECONTRIBUICAO Fisica Maria c2.txt
INSERECONTRIBUICAO Fisica Pedro c3.txt
```

**INSERELINK** Fisica Artes  
**IMPRIMEPAGINA** Fisica  
**INSERECONTRIBUICAO** Artes Maria c4.txt  
**INSERECONTRIBUICAO** Artes Ana c5.txt  
**RETIRACONTRIBUICAO** Artes Ana c5.txt  
**INSERELINK** Artes Fisica  
**INSERECONTRIBUICAO** UFES Joao c6.txt  
**INSERECONTRIBUICAO** UFES Joao c7.txt  
**INSERECONTRIBUICAO** UFES Ana c8.txt  
**INSERELINK** UFES Fisica  
**INSERELINK** UFES Artes  
**CAMINHO** UFES Fisica  
**CAMINHO** Fisica UFES  
**IMPRIMEWIKED**  
**FIM**

### Especificação dos Comandos

**INSEREPAGINA** <nome\_pagina><nome\_arquivo> : cria uma página WikED! com o nome e o arquivo especificados (devem ser únicos). Não é necessário criar o arquivo neste momento (isto será feito na impressão da página).

**RETIRAPAGINA** <nome\_pagina>: exclui a página da WikED!, excluindo, portanto, a lista de contribuições e links. Não é necessário excluir os arquivos de contribuições (apenas os nós das listas).

**INSEREEDITOR** <nome\_editor>: insere um editor com o nome especificado (deve ser único).

**INSERECONTRIBUICAO** <nome\_pagina><nome\_editor><nome\_arquivo>: insere uma contribuição de um dado editor para uma determinada página. O trecho de texto deve estar editado no arquivo especificado.

**RETIRACONTRIBUICAO** <nome\_pagina><nome\_editor><nome\_arquivo>: retira uma dada contribuição. Apenas o editor responsável pela contribuição pode retirá-la. O histórico da contribuição deve continuar ativo!

**INSERELINK** <pagina\_origem><pagina\_destino>: insere um link (nome do arquivo) para página destino, na página origem.

**RETIRALINK** <pagina\_origem><pagina\_destino>: retira um link (nome do arquivo) da página origem para uma página destino.

**CAMINHO** <pagina\_origem><pagina\_destino>: verifica se há caminho entre duas páginas (por meio das listas de links). Escreve no arquivo de log (HA/NAO HA CAMINHO DA <pagina\_origem> PARA <pagina\_destino>)

**IMPRIMEPAGINA** <nome\_pagina>: gera o arquivo e imprime as informações da página especificada.

**IMPRIMEWIKED** : gera os arquivos e imprime todas as informações das páginas da WikED!, como especificado acima.

**FIM**: determina a finalização do programa. Toda a memória alocada deve ser liberada.

### Dicas

- Ao percorrer as listas de links, cuidado para não entrar em um loop infinito (e.g., Física tem link para Artes e Artes tem link para Física).
- Caso o seu programa não consiga implementar algumas das funções acima, não deixe seu programa “quebrar”! Exiba uma mensagem no arquivo de log, dizendo “Função <nome da função> não suportada nesta versão da WikED!”. Assim, seu programa será avaliado de qualquer maneira, pelas funções que conseguiu implementar.
- Não esqueça de liberar toda memória alocada ao término da execução do programa!!!

Considerando o arquivo de entrada acima, espera-se o seguinte no arquivo de log:

### **Arquivo de log.txt**

ERRO: não existe a pagina Psicologia HA CAMINHO DA PAGINA UFES PARA Fisica NAO HA CAMINHO DA PAGINA Fisica PARA UFES
--

Exemplos de arquivos c1.txt, c2.txt, c3.txt, c4.txt, c5.txt, c7.txt, c8.txt, bem como dos arquivos fisica.txt, ufes.txt e artes.txt, serão disponibilizados pelo Classroom.

**BOM TRABALHO!!!**