

### Instruções:

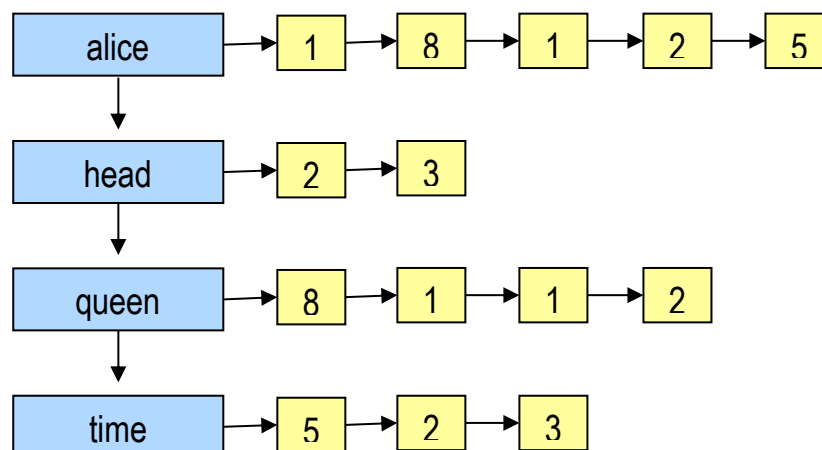
Este trabalho **pode ser feito em individualmente ou em dupla** e consiste no desenvolvimento de uma solução para geração automática de índice remissivo. Um índice remissivo indica em quais páginas uma determinada palavra está localizada.

Basicamente, é necessário realizar a leitura de um arquivo texto, e trabalhar com o armazenamento das palavras do texto em estruturas encadeadas para gerar o índice remissivo. A saída do programa consiste na apresentação dos resultados em modo texto.

Para usarmos arquivos de textos interessantes, sugerimos o acesso dos livros disponíveis no site do Projeto Gutenberg: <http://www.gutenberg.org/>. No Moodle já foram disponibilizados os arquivos de três livros disponíveis neste site, além de um arquivo de teste (veja o arquivo livros.zip). É importante observar que, para simplificar, foram retirados alguns comentários sobre o Projeto do início dos arquivos destes livros. Todos os livros estão em inglês, de maneira a facilitar a questão de tratamento dos acentos.

O primeiro passo para a realização desse trabalho é ler o arquivo e **obrigatoriamente criar estruturas encadeadas** capazes de guardar as palavras presentes no arquivo, de maneira que facilmente se possa montar o índice remissivo. O código para leitura do arquivo está disponível no Moodle, sendo necessário apenas integrá-lo à sua solução. É importante converter todas as palavras para minúsculas, para evitar duplicações desnecessárias.

Deve ser feita a modelagem de um conjunto de classes capaz de armazenar as palavras com as informações das páginas nas quais elas aparecem no arquivo texto original. Para facilitar, será considerado que a cada 40 linhas começa uma nova página. Veja na figura abaixo um exemplo dessa representação: as palavras estão na lista vertical, e cada uma possui uma lista das páginas onde ocorrem.



Qualquer sinal de pontuação deve ser removido no momento de armazenar a palavra. Lembre também que uma palavra pode aparecer mais de uma vez em uma página, mas no índice remissivo não pode aparecer o mesmo número de página duas vezes. Primeiro pense em como resolver este problema, para depois fazer a modelagem e desenvolver os algoritmos.

É importante lembrar que se deve projetar uma forma de armazenamento das palavras que permita um rápido acesso a cada palavra, uma vez que antes de incluir cada palavra deve-se verificar se a mesma já não foi armazenada, para apenas acrescentar mais um número de página onde ela aparece.

Ao ser iniciado, o programa deverá solicitar o nome do arquivo de texto a ser processado. Em seguida, deve apresentar as funcionalidades abaixo em um menu em modo texto para o usuário:

1. Exibir todo o índice remissivo (em ordem alfabética);
2. Pesquisar palavra (o usuário informa uma palavra; o sistema mostra as páginas em que a palavra ocorre).

#### **Observação:**

NÃO pode ser utilizada nenhuma classe da API Java de coleções, como *ArrayList*, *LinkedList*, *HashMap*, *HashSet*, etc. As implementações de estruturas encadeadas vistas em aula podem servir como base para as estruturas de dados utilizadas.

#### **Tarefas:**

- Implementar uma estrutura encadeada e algoritmos que permitam armazenar e consultar as informações do índice remissivo, conforme descrito;
- Modelar um conjunto de classes que permitam realizar as operações solicitadas;
- Implementar um programa que permita testar a estrutura de dados e os algoritmos implementados com os arquivos de exemplo fornecidos;

#### **Entrega:**

- Cada grupo deverá **entregar um arquivo zip contendo a implementação feita (todos os arquivos .java necessários para sua execução)**.
- Deve ser feito o *upload* deste arquivo através do *Moodle* **até a data e horário especificado**.

#### **Avaliação:**

Os seguintes critérios de avaliação serão utilizados:

- **Implementação da solução:** será averiguada se a solução usa estruturas encadeadas, está completa, eficiente e correta, e a qualidade e clareza do código implementado.

#### **Observações:**

- Trabalhos que se configurarem como CÓPIAS DAS SOLUÇÕES de outros colegas resultarão em NOTA ZERO para todos os alunos envolvidos.