








## Trabalho 02

Disciplina: Organização de Arquivos

Semestre: 2018/2





Entrega: 05/12/2018- 23h50.

### Correção:




-  Entregar arquivos fontes via website do curso ([aprender.unb.br](http://aprender.unb.br)) em único arquivo zipado.
-  Excepcionalmente para o Trabalho 2, não será exigido um relatório. Deverá ser entregue apenas código e comentários.
-  Incluir um arquivo de comentário ('instrucao.txt') com os nomes, matrículas e instruções para compilar o código.
-  Os trabalhos devem seguir boas práticas de programação (clareza, organização, comentários, etc.)
-  Trabalho identificado como cópia (qualquer uma das partes) receberá nota zero.

Grupos: Neste projeto está permitido a formação de grupos de no máximo dois alunos. Somente um dos membros do grupo deve entregar o trabalho onde conste nome e matrícula de todos os participantes.

### Especificações:

-  O arquivo `lista.txt` disponibilizado no moodle contém campos de tamanhos fixo com informação sobre NOME, MATRICULA, CURSO e TURMA nessa ordem. A partir desse arquivo, elabore um programa para criação de um arquivo de índices estruturado em árvore B, para armazenagem em memória secundária, com as seguintes características:
-  Índice primário - Utilize para criação da chave primária os campos de NOME e MATRICULA concatenando os 3 primeiros caracteres do nome, em maiúsculas, e 5 números de matrícula, num total de 8 caracteres.
-  Terminal de saída - Orie uma rotina para gravar em disco o arquivo de índices primários utilizando árvores B (`indicelista.bt`) e uma rotina para visualização do mesmo em monitor. Essa rotina deverá ser empregada após qualquer modificação sobre o arquivo. Modificações sobre o arquivo de dados também deverão ser gravadas em disco. Permita que o usuário escolha a ordem da árvore B.
-  Busca de registros - Orie uma rotina para buscar registros através de chave primária. A busca deve carregar em memória principal somente um nó da árvore B por vez. A rotina deve devolver o conteúdo do registro e o número de buscas (seeks) envolvidos.



-  Inclusão de registros - Orie uma rotina que permita a inclusão de novo registro no arquivo de dados e no arquivo de índices em árvore B. Utilizar técnicas de divisão e promoção para preservar a estrutura de árvore B. Não é necessária a re-utilização de espaço disponível (uso de LED) no arquivo de dados, a inclusão poderá ser feita ao final mesmo existindo espaço disponível.
-  Exclusão de registros - Orie uma rotina que permita a exclusão de registro (identificado por chave primária) do arquivo de dados e do arquivo de índices em árvore B. Apesar de não utilizarmos técnicas de re-utilização de espaço, marque com um símbolo os registros eliminados.
-  O programa deverá ser implementado em linguagem C ou C++. Em linguagem C, o programa deverá seguir o padrão ANSI. Em linguagem C++ o programa deverá ser compilável no sistema operacional UNIX.