



## Trabalho 01

**Disciplina:** Organização de Arquivos - turma C

**Semestre:** 2016/1

**Entrega:** 23/05/2016 - 23h50.

**Correção:**

- Entregar arquivos fontes via website do curso ([aprender.unb.br](http://aprender.unb.br)) em único arquivo zipado.
- Excepcionalmente para o Trabalho 1, não será exigido um relatório. Deverá ser entregue apenas código (em ANSI C ou C++) e comentários.
- Incluir um arquivo de comentário ('instrucao.txt') com os nomes, matriculas 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:**

- **Índice Primário** - A partir dos arquivos disponibilizados (*lista1.txt* e *lista2.txt* no [aprender.unb.br](http://aprender.unb.br)), crie um arquivo de índices primários simples (*indicelista1.ind* e *indicelista2.ind*). Utilize para a criação deste índice os campos MATRIC e NOME concatenados num total de 30 caracteres como chave primária (primeiro campo). O segundo campo (referência) deverá conter a informação necessária para identificar qual o registro do arquivo ao qual a chave se refere.

Exemplo do arquivo:

MATRIC	NOME	OP	CURSO	TURMA
024312	Leonardo Nunes de Souza	34	G	AB

- **Índice Secundário** - Crie dois arquivos de índices secundários, tendo como campos a coluna CURSO, correlacionados às suas respectivas chaves primárias. Estes arquivos de índices secundários deverão ser criados utilizando como princípio a técnica de Listas Invertidas, segundo apresentado em sala de aula.
- **Ordenação** - Para ordenar o arquivo de índice utilize um algoritmo de ordenação HEAPSORT seguindo o modelo da literatura existente. Crie uma rotina para a visualização do arquivo de índice (primário e secundário) ordenado no monitor.
- **Incluir Registros** - Deverá ser desenvolvido rotinas que permitam a inclusão de novos registros no arquivo de dados e também no arquivo de índices (primário e secundário). Lembrando que



o arquivo de índices deverá ser mantido ordenado. Mostrar no terminal de vídeo o arquivo de índice antes e depois da inclusão.

- **Excluir Registros** - Deverá ser desenvolvido rotinas que permitam a exclusão de registros do arquivo de dados e também do arquivo de índices (primário e secundário). Lembrando que o arquivo de índices deverá ser mantido ordenado e empregadas metodologias de re-utilização do espaço disponibilizado. Mostrar no terminal de vídeo o arquivo de índice antes e depois da exclusão.
- **Atualizar Registro** - Definir rotinas nas quais implementem a atualização de registros no arquivos de dados, e respectivamente no arquivos de índices (primário e secundário), seguindo as condições descritas abaixo:
  - Com mudança de chave primária
  - Sem mudança de chave primária
- **Intercalação (merge)** - Definir rotinas para intercalar os arquivos ordenados baseados em *lista1.txt* e *lista2.txt* gerando o arquivo *lista12.txt* intercalado por ordem de chave.
- Os resultados das rotinas e operações descritas acima para índices (primários e secundários) e arquivos devem ser apresentadas via monitor e gravados em disco.
- O programa deverá ser implementado em linguagem C ou C++. O programa deverá seguir o padrão ANSI e deverá ser obrigatoriamente compilável no sistema operacional UNIX.