

1º Trabalho Prático

O 1º trabalho prático relativo à disciplina **Organização e Recuperação de Dados** constitui-se do desenvolvimento de um programa conforme as especificações abaixo.

O programa deverá ser desenvolvido na linguagem C (compilador gcc) e deverá ser feito individualmente ou em duplas.

A entrega do trabalho será feita pelo *moodle* e deverão ser entregues como parte do trabalho:

- relatório (.pdf) de uma página contendo informações sobre: S.O., compilador e IDE utilizados no desenvolvimento, bem como outras decisões de projeto e configurações específicas necessárias para o correto funcionamento do programa;
- código-fonte do programa (arquivos .c e .h (se houver)).

Especificação

O arquivo ***dados-inline.txt*** (disponível no *moodle*) possui informações a respeito de candidatos ao vestibular. O programa deverá ser capaz de importar os dados do arquivo ***dados-inline.txt*** e armazená-los na forma de registros de tamanho variável. Inicie cada registro indicando seu tamanho em bytes e utilize o “|” como separador de campos. Os campos dos registros são os seguintes:

1. INSCRIÇÃO: Número de inscrição do candidato;
2. NOME: Nome completo do candidato;
3. CURSO: Nome do curso no qual está inscrito;
4. SCORE: *Score* final obtido no vestibular.

O programa deverá oferecer as seguintes funcionalidades:

- Importação dos dados;
- Busca de registro pelo número de INSCRIÇÃO;
- Inserção de novo registro;
- Remoção de registro.

Deverá ser exibida uma interface na qual se poderá escolher entre as funcionalidades do programa. A importação poderá ser feita em qualquer momento da execução. Sempre que ativada, essa opção importará os dados do arquivo ***dados-inline.txt*** e criará o **arquivo de registros**. As alterações que venham a ocorrer no arquivo de registros após a importação (i.e., inserções e remoções) deverão ser persistentes. Dessa forma, sempre que o programa for iniciado, ele deverá abrir o arquivo de registros, se o mesmo existir, permitindo assim a sua edição. Caso contrário, o arquivo de registros deverá ser criado a partir de uma importação. Durante a execução do programa, o arquivo de registros deverá ser mantido e acessado em disco.

A busca de registros será feita pela INSCRIÇÃO e será sequencial. Uma vez que o registro buscado for encontrado, seus campos deverão ser mostrados na tela, um por linha.

A remoção de registros com um determinado número de INSCRIÇÃO será lógica e o espaço disponível resultante da remoção deverá ser inserido na Lista de Espaços Disponíveis (LED). **A LED deverá ser mantida no próprio arquivo de registros**, conforme visto em aula, e o programa deverá implementar todos os mecanismos necessários para o seu gerenciamento e para a reutilização dos espaços disponíveis. Adote o **primeiro ajuste** (*first fit*) como estratégia de gerenciamento da LED. **Questões práticas:** (1) use o * seguido de um | para marcar registros removidos e grave o ponteiro da LED também seguido de um |. Isso facilitará a leitura desses campos posteriormente. (2) Use a função *fscanf()* (biblioteca *stdio.h*) para gravar o ponteiro da LED como uma *string*. Isso facilitará a visualização da LED como parte do arquivo gerado, bem como a verificação dos ponteiros.

A inserção de novos registros se dará pela digitação dos valores dos campos pelo usuário. No momento da inserção do novo registro no arquivo, a LED deverá ser consultada. Se existir um espaço disponível adequado para a inserção, o novo registro deverá ser inserido nesse espaço. Sobras de espaço resultante da inserção poderão ser mantidas como parte do registro. Se na LED não houver um espaço disponível adequado para o novo registro, a inserção deverá ser feita no final do arquivo.