

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).

O trabalho poderá ser feito em duplas.

Deverão ser entregues como parte do trabalho:

- Relatório (.pdf) contendo informações relativas ao desenvolvimento do programa, como decisões de projeto, configurações específicas para a compilação do código (caso exista), etc;
- Código-fonte do programa (arquivos .c e possíveis .h).

A entrega do trabalho deverá ser feita exclusivamente via *Moodle*.

Cadastro de cães

O objetivo deste trabalho é exercitar a parte prática referente à criação e manutenção de arquivos de registros em C, bem como o uso de índices lineares e listas invertidas.

Deverá ser criado um programa para gerenciar um cadastro de cães. Esse cadastro manterá informações referentes à raça, grupo da raça e dados do indivíduo. Para isso serão criados dois arquivos de dados: um de raças e um de indivíduos. Também serão criados os arquivos necessários para a manutenção de índices conforme especificado mais adiante.

O arquivo de raças conterá os dados referentes às raças de cães. Cada raça tem um identificador único (ID-R) e faz parte de um único grupo. Cada grupo também tem um identificador único (ID-G) e pode abranger várias raças. Para cada uma das raças cadastradas haverá um registro com os seguintes campos:

- ID-R (identificador de raça)
- RAÇA (nome da raça)
- ID-G (identificador de grupo)
- GRUPO (nome do grupo)

O arquivo de indivíduos conterá os dados referentes aos indivíduos, i.e., aos cães propriamente ditos. Cada indivíduo tem um identificador único (ID-I) e possui uma única raça (ID-R). Para cada um dos indivíduos cadastrados haverá um registro com os seguintes campos:

- ID-I (identificador do indivíduo)
- ID-R (identificador da raça)
- NOME (nome do indivíduo)
- SEXO (sexo)

Tanto os registros do arquivo de raças quanto os do arquivo de indivíduos deverão estar no mesmo formato de registro utilizado nas aulas práticas de laboratório (registros de tamanho variável com um campo identificador de tamanho no início do registro).

O programa deverá ser capaz de importar tanto os dados do arquivo de raças quanto do arquivo de indivíduos, desde que esses dados estejam disponíveis em arquivos texto contendo um valor de campo por linha. Sempre que uma importação ocorrer, o arquivo antigo será substituído pelo novo contendo os dados importados. Na prática, a importação transforma os dados disponíveis em um formato (um campo por linha) em outro (registros de tamanho variável com identificador de tamanho). Dados para a importação do arquivo de raças e indivíduos estão disponíveis no *moodle* junto com esta especificação.

Além da importação dos arquivos de dados, o programa deverá criar e manter um sistema de indexação baseado em índices lineares (primários e secundários) e em listas invertidas.

O arquivo de raças será indexado por um índice primário por ID-R (ip1) e um índice secundário por ID-G (is2). O ip1 deverá ser implementado como um índice linear com uma lista invertida de ID-I (indivíduos). O is2 também deverá ser implementado como um índice linear com uma lista invertida de ID-R (raças). Cada lista invertida será um arquivo sequencial de registros de tamanho fixo.

O arquivo de indivíduos será indexado por um índice primário por ID-I (ip3). O ip3 também deverá ser implementado como um índice linear.

Todos os arquivos de índice devem ser automaticamente construídos sempre que houver uma importação de dados. **Durante a execução do programa, os índices deverão ser mantidos e acessados em memória, enquanto os arquivos de dados deverão ser mantidos e acessados em disco.** Os arquivos de índice poderão ser atualizados em disco quando o programa for encerrado.

O programa deverá oferecer as seguintes funcionalidades ao usuário:

- Importação de dados
- Inserção de um novo indivíduo
- Buscas

A importação poderá ser feita em qualquer momento da execução do programa. Sempre que ativada, essa opção limpará o arquivo antigo e importará os dados de um arquivo de entrada. A importação deve ser capaz de importar qualquer arquivo que esteja no mesmo formato descrito no início desta especificação. Note que toda vez que for feita a importação de dados será preciso reconstruir os índices e atualizá-los em disco.

A inserção de novos registros de indivíduos se dará pela digitação dos valores dos campos pelo usuário. Assim que for feita a entrada do valor do campo ID-I, o programa deverá se certificar de que o valor digitado não existe no arquivo, uma vez que números de ID-I duplicados não são permitidos. Caso o número digitado já exista, o usuário deve ser avisado e um novo valor de ID-I deverá ser solicitado. Após a validação do ID-I, a entrada do restante dos campos deverá prosseguir normalmente. O valor do campo ID-R também deverá ser validado. Todo novo registro deverá ser inserido no final do arquivo.

Deverá ser possível realizar buscas usando os três índices. As seguintes buscas devem ser implementadas:

- Busca de um registro de raça pelo ID-R (ip1)
- Busca de uma lista de raças pelo ID-G (is2+listaInvertida)
- Busca de um registro de indivíduo pelo ID-I (ip3)
- Busca de uma lista de indivíduos pelo ID-R (ip1+listaInvertida)
- Busca de uma lista de indivíduos pelo ID-R e pelo SEXO (ip1+listaInvertida)

Deverá ser exibida ao usuário uma interface na qual ele poderá escolher entre as diversas funcionalidades do programa. Todas as funcionalidades descritas devem estar entre as opções disponibilizadas. Aspectos de interface não explicitados neste documento deverão ser decididos pelos membros da dupla ou consultados com a profa.

A dupla deverá tomar todas as decisões de projeto não explicitadas neste documento, sempre mediante justificativa. Por decisões de projeto entende-se as escolhas feitas pela equipe que modificam aspectos de implementação em situações em que mais de uma forma de implementação poderia ser escolhida. Outras dúvidas poderão ser esclarecidas com a profa.

BOM TRABALHO!