

Exercício

O objetivo do exercício é gerenciar um sistema de histórico escolar “limpo” de uma dada instituição. O sistema armazena as seguintes informações: ID do aluno, sigla da disciplina, nome do aluno, nome da disciplina, média e frequência. A chave primária é composta pela composição “ID do aluno+Sigla da disciplina”. O arquivo a ser criado deve ser de registros e campos de tamanho variável, com um inteiro (4 bytes) no início do registro indicando o tamanho do registro, e com campos separados pelo caractere ‘#’.

ID do aluno	Sigla da disciplina	Nome do aluno	Nome da disciplina	Média	Frequência
3 caracteres (fixo)	3 caracteres (fixo)	50 caracteres (máximo)	50 caracteres (máximo)	float (a ser bufferizado)	float (a ser bufferizado)

Ex.: 53001#ED2#Paulo da Silva#Estruturas de Dados 2#7.3#75.4

As seguintes operações deverão estar disponíveis:

1. Inserção
2. Procurar por “ID do aluno+Sigla da disciplina” (índice primário)
3. Procurar por “Nome do aluno” (índice secundário / lista invertida)
4. Carrega Arquivos (opcional)

Inserção (1)

Insere o registro no final do arquivo. Os dados a serem inseridos devem ser recuperados de um arquivo a ser fornecido no momento da execução do programa (vide Opção 4).

Pesquisa por chave primária (2)

Vocês devem criar um arquivo de índice que contenha a lista dos “ID do aluno+Sigla da disciplina” presentes em seu arquivo de dados junto com o deslocamento necessário para acessar o registro de cada chave presente no arquivo. Assim, uma consulta deve primeiramente procurar a chave desejada neste novo arquivo e depois acessar diretamente o registro desejado no arquivo de dados. Os dados relacionados ao “ID do aluno+Sigla da disciplina” pesquisado devem ser exibidos.

Observações:

- (1) Agora, a inserção de um registro requer a manipulação de 2 arquivos (dados e índice).
- (2) A busca por um código no índice primário pode ser feita sequencialmente ou por pesquisa binária.
- (3) O índice deve ser mantido em memória principal e, em caso do programa ser interrompido inesperadamente, o índice deve ser recriado a partir do arquivo de dados. Desse modo, deve existir uma função que carrega o índice para a memória e uma que recria o índice quando necessário. Para criar/recriar o índice utilizem o Keysorting.

Pesquisa por chave secundária (3)

Construa um índice secundário para acesso pela “Nome do aluno”. Este índice consiste de 2 arquivos, formando uma lista invertida com o índice primário e o arquivo de dados. O primeiro arquivo do índice secundário contém a lista dos nomes dos alunos, sem repetições, presentes no arquivo de dados junto com o deslocamento necessário para acessar uma lista de registros de cada nome no segundo arquivo.

nome_1 offset1 nome_2 offset2 ...

Já o segundo arquivo do índice secundário contém uma chave primária (“ID do aluno+Sigla da disciplina”) correspondente a um aluno seguida de um novo deslocamento para este mesmo arquivo, formando uma lista ligada de chaves primárias. Quando este deslocamento for -1 quer dizer que a lista ligada acabou.

ID+Sigla offset1 ID+Sigla offset2 ...

Observações: as mesmas da Opção 2.

Carrega Arquivos (4)

A fim de facilitar os testes, serão fornecidos três arquivos: (a) “insere.bin”, (b) “busca_p.bin” e (c) “busca_s.bin”. O primeiro (a) conterá os dados a serem inseridos durante os testes (não necessariamente todos os dados serão inseridos). Para tanto, uma sugestão é carregar o arquivo em memória (um vetor de struct) e ir acessando cada posição conforme as inserções vão ocorrendo. Note que é possível encerrar a execução e recomendar a execução, sendo necessário marcar, de algum modo, quantos registros já foram utilizados do mesmo.

Em relação a (b), o arquivo conterá uma lista de “ID do aluno+Sigla da disciplina” a serem utilizados durante a pesquisa por chave primária (opção 2). A ideia é a mesma já descrita, ou seja, carregar o arquivo em memória (um vetor de struct) e ir acessando cada posição conforme as buscas vão ocorrendo. Note que é possível encerrar a execução e recomendar a execução, sendo necessário marcar, de algum modo, quantos registros já foram utilizados do mesmo. Em relação a (c), considere as mesmas observações em relação ao item (b); porém, neste caso, o arquivo conterá uma lista de “alunos” a serem utilizados durante a pesquisa por chave secundária (opção 3).

Observações Gerais:

- (1) Não criar os arquivos toda vez que o programa for aberto (fazer verificação).
- (2) O arquivo principal deve ser manipulado totalmente em memória secundária!