

Trabalho Prático 02: Manipulação e Reorganização de Arquivos de Dados

Vivian Cristina Santana de Souza Lopes

Disciplina: Algoritmos e Estruturas de Dados II

Professor: Rafael Alexandre

1. Introdução

Ao longo da vida útil de um arquivo de dados, diversas operações de manutenção — como inserção, exclusão e atualização de registros — modificam continuamente a forma como as informações são distribuídas nos blocos de armazenamento. Essas alterações, embora necessárias para garantir a persistência e a integridade dos dados, acabam gerando efeitos colaterais que comprometem a eficiência do sistema. Entre os principais problemas estão a fragmentação, o desperdício de espaço e a perda de desempenho no acesso às informações.

A fragmentação interna ocorre quando parte do espaço de um bloco permanece inutilizada, seja pelo tamanho fixo dos registros ou pelo desalinhamento entre o tamanho do registro e o tamanho do bloco. Já a fragmentação externa manifesta-se quando pequenas áreas livres ficam espalhadas entre os blocos, resultado de exclusões ou realocações, dificultando o armazenamento de novos registros de forma contígua. Esses fenômenos evidenciam a importância da gestão de espaço livre, que consiste em controlar e reaproveitar as áreas disponíveis para garantir que novos registros sejam alocados de maneira eficiente.

Para mitigar os efeitos da fragmentação, torna-se necessária a reorganização física do arquivo, um processo que envolve a compactação dos dados e a redistribuição dos registros nos blocos. Essa prática não apenas reduz o desperdício de espaço, mas também melhora significativamente o desempenho das operações de leitura e escrita, assegurando maior eficiência no armazenamento.

Este trabalho dá continuidade ao Trabalho Prático I, no qual foi desenvolvida a simulação da persistência de registros de alunos em blocos de tamanho fixo ou variável. Agora, no Trabalho Prático II, o foco recai sobre a manipulação dinâmica dos registros e a manutenção da estrutura física do arquivo. O arquivo utilizado, denominado `alunos.dat`, armazena registros de forma persistente em disco e serve como base para a implementação de mecanismos de exclusão lógica, reaproveitamento de áreas livres, reorganização física e geração de relatórios de eficiência.

2. Objetivos

O objetivo desse trabalho é aprofundar os conceitos de armazenamento em blocos, abordados na disciplina de AEDS II, por meio da prática de técnicas que simulam situações reais de gerenciamento de arquivos. Dessa forma, busca-se não apenas

compreender os impactos da fragmentação e da gestão de espaço livre, mas também propor soluções que tornem o acesso às informações mais rápido e eficiente.

Os principais objetivos do Trabalho Prático II são:

- Implementar operações de inserção, edição e exclusão de registros em um arquivo estruturado em blocos.
- Controlar o reaproveitamento de espaços livres oriundos de exclusões lógicas.
- Garantir que todas as operações respeitem o tamanho máximo de bloco definido pelo usuário.
- Permitir a inclusão de novos registros tanto em blocos parcialmente ocupados quanto no final do arquivo.
- Implementar uma rotina de reorganização física, eliminando fragmentações internas e externas.
- Gerar relatórios comparativos de eficiência antes e depois da reorganização do arquivo.

3. Metodologia

Antes da execução das operações principais, o sistema permite que o usuário gere um conjunto de registros fictícios, definindo a quantidade total de alunos a serem criados.

Essa funcionalidade foi integrada diretamente ao fluxo do programa e é essencial para:

- Realizar testes em larga escala;
- Avaliar o comportamento do armazenamento;
- Produzir mapas de ocupação e relatórios significativos.

Os dados fictícios são gerados de forma determinística, garantindo consistência nos testes.

O campo removido é utilizado para implementar a remoção lógica, permitindo que o espaço ocupado pelo registro seja reutilizado futuramente sem a necessidade de reorganizar imediatamente o arquivo.

Campo	Matricula	Tipo
Matricula	Inteiro (9 dígitos)	Identificador unico aluno
Nome	String (até 50 caracteres)	Nome do aluno
Cpf	String (11 caracteres)	CPF do aluno
Curso	String (até 30 caracteres)	Curso em que o aluno está matriculado

Filiação Mãe	String (até 30 caracteres)	Nome completo da mãe
Filiação Pai	String (até 30 caracteres)	Nome completo do pai
Ano de ingresso	Inteiro (4 dígitos)	Ano de entrada na instituição
CA	Float (2 casas decimais)	Coeficiente Acadêmico

O usuário define, no início da execução:

O tamanho máximo do bloco, em bytes;

O tipo de registro:

1. Fixo: todos os registros ocupam o mesmo espaço;
2. Variável: o tamanho depende do conteúdo textual armazenado.

Essa flexibilidade permite comparar o impacto dos diferentes tipos de organização na eficiência do armazenamento.

Cada registro, ao ser inserido, tem seu tamanho calculado conforme o tipo selecionado. O sistema então organiza os registros sequencialmente dentro dos blocos, respeitando o limite máximo de bytes por bloco.

Sempre que um registro não cabe no bloco atual, um novo bloco é criado, e os metadados de ocupação são atualizados. Esses metadados são armazenados em uma estrutura auxiliar que mantém:

1. Identificador do bloco;
2. Quantidade de bytes utilizados.

Essa abordagem permite simular o funcionamento de armazenamento em disco.

4. Operações Implementadas

4.1 Inserção de Registros

A operação de inserção segue a seguinte estratégia:

Inserir novos registros no arquivo.

Prioriza o reaproveitamento de espaços deixados por registros removidos.

Caso não haja espaço disponível, o registro é inserido no final do arquivo.

Essa estratégia reduz a fragmentação externa e melhora a eficiência do armazenamento.

4.2 Edição de Registros

A edição permite a alteração de qualquer campo do registro de um aluno, identificado pela matrícula.

Permite alterar campos do registro.

Se o novo tamanho for menor ou igual ao original, a edição ocorre no mesmo local.

Se o novo tamanho for maior, o registro antigo é marcado como removido e um novo registro é inserido, caracterizando realocação.

4.3 Exclusão de Registros

A exclusão é realizada de forma lógica, marcando o campo removido com valor 1.

O registro permanece fisicamente no arquivo, mas passa a ser ignorado nas listagens e pode ser reutilizado em futuras inserções.

Essa abordagem evita custos elevados de escrita em disco a cada exclusão.

4.4 Listagem de Registros

A listagem apresenta apenas os registros ativos, exibindo informações como matrícula, nome, curso, ano de ingresso e coeficiente de rendimento.

4.5 Mapa de Ocupação dos Blocos

O programa gera um relatório que apresenta:

1. Número total de blocos.
2. Quantidade de registros ativos e removidos.
3. Bytes ocupados e livres por bloco.
4. Ocupação média e eficiência total do arquivo.

Esse mapa permite visualizar claramente o impacto das operações realizadas, especialmente inserções, edições e exclusões.

4.6 Reorganização do Arquivo

A reorganização física cria um novo arquivo (alunos_reorg.dat) contendo apenas registros ativos, eliminando espaços desperdiçados por exclusões lógicas.

Durante esse processo, o sistema calcula e apresenta:

1. Quantidade de blocos antes e depois;
2. Ocupação média dos blocos;
3. Eficiência total de armazenamento;
4. Ganho de eficiência;
5. Número de blocos liberados.

Esse procedimento simula operações reais de compactação de arquivos utilizadas em bancos de dados.

A eficiência de armazenamento é calculada pela relação entre:

$$\text{Eficiência} = \text{Bytes Utilizados} / \text{total de Bytes Disponível} \times 100.$$

Os resultados são apresentados de forma clara e estruturada, permitindo análise comparativa antes e depois da reorganização.

5. Casos de Teste Realizados

Todos os testes estão anexados ao trabalho, como teste 1, teste 2, teste 3, teste 4, teste 5, teste 6 e apresentaram resultados compatíveis com o esperado.

Observou-se que o uso de registros de tamanho variável proporciona maior flexibilidade e melhor aproveitamento dos blocos, enquanto os registros fixos apresentam simplicidade de implementação, porém maior fragmentação interna.

Em cenários com grande volume de registros e tamanho de bloco adequado, observa-se elevada taxa de ocupação dos blocos, resultando em mapas de ocupação extensos e com pouca variação percentual entre blocos.

O mapa de ocupação evidenciou que a maioria dos blocos foi totalmente preenchida, com exceção do último bloco, caracterizando fragmentação interna típica do uso de registros de tamanho fixo.

A exclusão lógica foi realizada com sucesso, mantendo o registro fisicamente no arquivo e liberando o espaço para possível reaproveitamento futuro. Observou-se o reaproveitamento de espaços anteriormente liberados por exclusões lógicas, evitando a criação de novos blocos e aumentando a eficiência do armazenamento.

O ganho reduzido de eficiência após a reorganização deve-se ao baixo número de registros removidos no cenário de teste, evidenciando que a eficácia da reorganização está diretamente relacionada ao grau de fragmentação existente no arquivo.

6. Conclusão

O trabalho atingiu todos os objetivos propostos, implementando com sucesso um sistema completo de armazenamento em blocos com suporte a registros fixos e variáveis. As funcionalidades de inserção, edição, exclusão, reorganização física e geração de relatórios permitiram analisar, na prática, os impactos das decisões de organização de arquivos. A simulação desenvolvida reforça conceitos fundamentais da disciplina de Algoritmos e Estruturas de Dados II, proporcionando uma visão concreta de como sistemas de armazenamento gerenciam espaço e eficiência.