

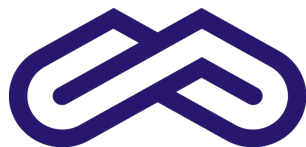
UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
CURSO DE SISTEMAS DE INFORMAÇÃO

MARCOS LEANDRO PINTO MOURA

**DOCUMENTAÇÃO DO TRABALHO PRÁTICO
(AEDS-I)**

MONTES CLAROS / MG

JULHO DE 2023



SUMÁRIO

1. INTRODUÇÃO	2
2. IMPLEMENTAÇÃO	4
3. CONCLUSÃO	9
4. BIBLIOGRAFIA	10

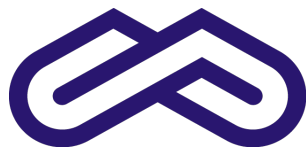
1. INTRODUÇÃO

O sistema de academia é um programa em C que permite cadastrar, listar, pesquisar, excluir e alterar dados de alunos em um arquivo de texto. O programa foi desenvolvido para auxiliar na gestão de alunos em um contexto educacional, por exemplo, em uma academia, escola associação etc.

O objetivo do programa é fornecer um sistema simples e eficiente para o gerenciamento dos alunos, permitindo o registro de informações como código único do aluno, nome, sexo, endereço, peso, altura, curso e mensalidade. Além disso, o programa realiza cálculos como o Índice de Massa Corporal (IMC), o peso ideal dos alunos e o somatório total das mensalidades por curso juntamente com o total geral.

A seguir, será fornecida uma visão geral sobre o funcionamento do programa, descrevendo as principais funcionalidades e como elas são implementadas.

Visão geral do funcionamento do programa: O programa apresenta um menu com várias opções para o usuário escolher. Cada opção corresponde a uma função específica do programa. As funcionalidades principais são:

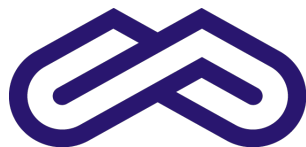


UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

CURSO DE SISTEMAS DE INFORMAÇÃO

- 1) Cadastrar Aluno: Permite ao usuário inserir os dados de um novo aluno, incluindo código, nome, sexo, endereço, peso, altura, curso e mensalidade. Os dados são armazenados em um arquivo de texto chamado "aluno.txt".
- 2) Pesquisar Aluno: Permite ao usuário pesquisar um aluno pelo código. O programa percorre o arquivo "aluno.txt" e exibe as informações do aluno correspondente, se encontrado.
- 3) Alterar Aluno: Permite ao usuário alterar por completo (exceto o código) o cadastro de um aluno a partir do código. Assim que digitado, a ficha do aluno é imprimida para conferir, após isso, abre a tela e a função para recadastrar esse aluno, sobrescrevendo o aluno anterior no arquivo "aluno.txt". Essa função é útil quando ocorrem alterações nas informações do aluno, como mudança de endereço ou atualização dos dados pessoais.
- 4) Excluir Aluno: Permite ao usuário remover um aluno do arquivo "alunos.txt" com base no nome fornecido. O programa cria um arquivo temporário, copia todos os alunos, exceto o aluno a ser excluído, para o arquivo temporário e, em seguida, renomeia o arquivo temporário como "alunos.txt".
- 5) Listar Alunos: Mostra na tela todos os alunos cadastrados no arquivo "aluno.txt", exibindo suas informações principais como código e nome.
- 6) Listar Alunos com IMC > 30: Mostra na tela os alunos que possuem Índice de Massa Corporal (IMC) maior que 30. O programa calcula o IMC com base no peso e altura dos alunos.



UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

CURSO DE SISTEMAS DE INFORMAÇÃO

- 7) Listar Alunos com Peso Atual e Peso Ideal: Exibe na tela o peso atual e o peso ideal de cada aluno. O peso ideal é calculado com base no sexo e altura dos alunos.
- 8) Relatório dos Cursos: Realiza uma análise dos dados dos alunos cadastrados, somando as mensalidades por curso e o total. Isso possibilita obter um relatório financeiro por curso, auxiliando na gestão e no acompanhamento das receitas geradas por cada modalidade de ensino.

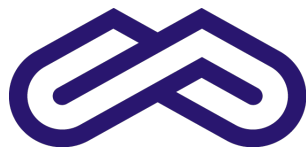
O programa continua executando até que o usuário escolha a opção de “FECHAR”, que possui um “break” para finalizar o programa.

2. IMPLEMENTAÇÃO

Estrutura de Dados: O programa utiliza a estrutura de dados “Aluno” para armazenar as informações de cada aluno. Essa estrutura possui os seguintes campos: código, nome, endereço, sexo, altura, peso, curso, mensalidade.

Funcionamento das principais funções:

- 1) Cadastrar Aluno:
 - O usuário fornece as informações do aluno, código, nome, endereço, sexo, altura, peso, curso, mensalidade
 - Os dados são validados e armazenados em uma variável do tipo “Aluno”.
 - O arquivo "aluno.txt" é aberto em modo de escrita binária (wb).
 - Os dados do aluno são escritos no arquivo usando a função “fwrite”.
 - O arquivo é fechado.



UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

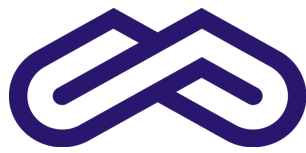
CURSO DE SISTEMAS DE INFORMAÇÃO

2) Pesquisar Aluno:

- O usuário fornece o nome do aluno a ser pesquisado.
- O arquivo "aluno.txt" é aberto em modo de leitura binária (rb).
- O programa percorre o arquivo usando um loop while e lê os dados de cada aluno usando a função "fread".
- Se o nome do aluno lido for igual ao nome fornecido, os dados do aluno são exibidos na tela.
- O arquivo é fechado.

3) Alterar Aluno:

- O usuário fornece o código do aluno a ser alterado.
- O arquivo "aluno.txt" é aberto em modo de leitura e escrita binária (rb+).
- O programa percorre o arquivo usando um loop "while" e lê os dados de cada aluno usando a função "fread".
- Se o código do aluno lido for igual ao código fornecido pelo usuário, o programa solicita as novas informações para cada campo, nome, endereço, sexo, altura, peso, curso, mensalidade.
- Os dados são validados e armazenados em variáveis do tipo "Aluno".
- A posição atual do ponteiro de arquivo é ajustada para a posição correta usando a função "fseek".
- Os novos dados do aluno são escritos no arquivo usando a função "fwrite".
- O arquivo é fechado.



UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

CURSO DE SISTEMAS DE INFORMAÇÃO

4) Excluir Aluno:

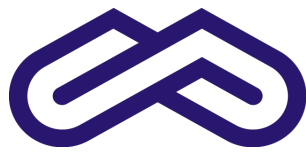
- O usuário fornece o nome do aluno a ser excluído.
- O arquivo "aluno.txt" é aberto em modo de leitura binária (rb).
- Um novo arquivo temporário, "temp.txt", é aberto em modo de escrita binária (wb).
- O programa percorre o arquivo original usando um loop "while" e lê os dados de cada aluno usando a função "fread".
- Se o nome do aluno lido for diferente do nome do aluno a ser excluído, os dados são escritos no arquivo temporário usando a função "fwrite".
- Após percorrer todo o arquivo, o arquivo original é fechado e removido usando a função "remove".
- O arquivo temporário é renomeado para "aluno.txt" usando a função "rename".
- Uma mensagem de confirmação ou erro é exibida na tela.

5) Listar Alunos:

- O arquivo "aluno.txt" é aberto em modo de leitura binária (rb).
- O programa percorre o arquivo usando um loop "while" e lê os dados de cada aluno usando a função "fread".
- Os dados do aluno são exibidos na tela.
- O arquivo é fechado.

6) Listar Alunos com IMC > 30:

- O arquivo "aluno.txt" é aberto em modo de leitura binária (rb).
- O programa percorre o arquivo usando um loop "while" e lê os dados de cada aluno usando a função "fread".
- O programa calcula o IMC do aluno com base no peso e altura fornecidos.
- Se o IMC for maior que 30, os dados do aluno são exibidos na tela.
- O arquivo é fechado.



UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

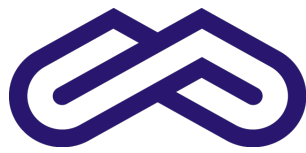
CURSO DE SISTEMAS DE INFORMAÇÃO

7) Listar Alunos com Peso Atual e Peso Ideal:

- O arquivo "aluno.txt" é aberto em modo de leitura binária (rb).
- O programa percorre o arquivo usando um loop "while" e lê os dados de cada aluno usando a função "fread".
- O programa calcula o peso ideal do aluno com base no sexo e altura fornecidos.
- Os dados do aluno, incluindo peso atual e peso ideal, são exibidos na tela.
- O arquivo é fechado.

8) Relatório dos Cursos:

- O arquivo "aluno.txt" é aberto em modo de leitura binária (rb).
- O programa percorre o arquivo usando um loop "while" e lê os dados de cada aluno usando a função "fread".
- Para cada aluno lido, é feita a verificação do curso em que ele está matriculado.
- A mensalidade do aluno é adicionada à soma correspondente ao curso.
- As mensalidades de todos os cursos e alunos são somadas, formando o valor total arrecadado pela academia.
- Ao final do arquivo, são exibidas as somas das mensalidades por curso.
- O arquivo é fechado.



UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

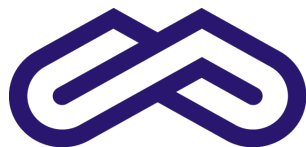
CURSO DE SISTEMAS DE INFORMAÇÃO

⇒ Formato das entradas e saídas de dados:

- a. O programa solicita informações ao usuário por meio de mensagens exibidas na tela.
- b. As entradas do usuário são feitas por meio do teclado usando a função “gets” para ler strings e “scanf” para ler valores numéricos.
- c. As saídas do programa são exibidas na tela por meio de mensagens de texto.

⇒ Decisões tomadas:

- a. Utilizei um arquivo de texto ("aluno.txt") para armazenar os dados dos alunos, em que cada linha representa um aluno com suas respectivas informações.
- b. Optei por usar a função “fwrite” e “fread” para escrever e ler os dados dos alunos em formato binário. Isso permite uma leitura e gravação mais eficiente dos dados.
- c. As funções foram implementadas com base nas especificações fornecidas, utilizando as estruturas de controle adequadas (loops, condições etc.) para realizar as operações necessárias nos arquivos e manipulação dos dados dos alunos.



3. CONCLUSÃO

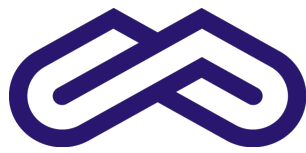
Ao longo do processo de implementação, enfrentei algumas dificuldades, como a manipulação correta de arquivos binários, a validação dos dados fornecidos pelo usuário e a lógica envolvida em algumas operações, como cálculos de IMC e pesos ideais. Além disso, a implementação de cada nova funcionalidade exigiu um cuidado extra para garantir a integridade dos dados e o correto funcionamento do programa.

O tempo necessário para a conclusão do programa foi maior do que o esperado, devido às complexidades envolvidas e às dificuldades encontradas. No entanto, com perseverança e dedicação, consegui superar os obstáculos e implementar todas as funcionalidades de forma eficaz.

O programa finalizado apresenta um conjunto de funcionalidades completas, que permitem ao usuário cadastrar, manipular e visualizar os dados dos alunos de forma organizada. Ele oferece recursos importantes para a gestão e o controle das informações dos alunos, auxiliando na administração dos cursos oferecidos.

É importante ressaltar que, mesmo após a conclusão do programa, sempre há espaço para melhorias e otimizações. Porém, com a implementação realizada, o sistema de cadastro de alunos se mostra eficaz e capaz de atender às necessidades propostas.

No geral, este trabalho representou um desafio interessante e enriquecedor, permitindo o aprimoramento das habilidades em programação e a compreensão da aplicação prática dos conceitos estudados. A superação das dificuldades encontradas ao longo do desenvolvimento do programa foi recompensadora e contribuiu para o crescimento profissional.



Unimontes
Universidade Estadual de Montes Claros

UNIVERSIDADE ESTADUAL DE MONTES CLAROS – UNIMONTES

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

CURSO DE SISTEMAS DE INFORMAÇÃO

Estou satisfeito com o resultado alcançado e confiante de que o programa atende aos requisitos estabelecidos, fornecendo um sistema funcional e eficiente para o gerenciamento de alunos em cursos.

4. BIBLIOGRAFIA

"UFPR." <https://www.inf.ufpr.br/lesoliveira/download/c-completo-total.pdf>.

"DevDocs — C documentation." <https://devdocs.io/c/>.