



Universidade Federal de Uberlândia
Faculdade de Computação

Trabalho de Análise e Estrutura de Dados I

Guilherme Pacheco de Oliveira - 11411BCC051

Descrição do Problema: Cadastro de Alunos

O problema em questão trata-se de um projeto em menor escala de um sistema de cadastros de uma faculdade.

A solução deve oferecer as seguintes opções:

- Ler um arquivo “alunos.txt” com os nomes dos alunos de um curso e com suas respectivas matrículas e armazenar esses dados em ordem alfabética pelo nome em uma estrutura de dados.
- Ler um arquivo “disciplinas.txt” com o nome da disciplina, seu código e os créditos correspondentes e armazenar esses dados em ordem alfabética pelo nome em outra estrutura de dados.
- Para cada aluno matriculado, devem ter as informações: Disciplinas Concluídas e Aprovadas com a nota final, Disciplinas sendo cursadas com a nota parcial e Disciplinas Pendentes ou Reprovadas, com a nota final.
- Cadastro, edição e remoção de alunos.
- Na opção de edição, deve ser possível inserir disciplinas em andamento (cada aluno pode cursar apenas 6 matérias), inserir notas para as disciplinas em andamento e encerrar uma disciplina.
- No caso de encerramento de uma disciplina, deve constatar se o aluno foi aprovado ou reprovado
- Inserir uma disciplina reprovada para a lista de disciplinas cursando, constando de algum modo que a disciplina está sendo cursada novamente, e se o aluno já tiver cursado a disciplina 4 vezes, ela não poderá ser cursada novamente.
- O programa deve atualizar os arquivos “alunos.txt” e “disciplinas.txt”, salvando as listas no arquivo

Diretórios e arquivos do Programa:

Diretório Geral:

```
cadastro_alunos/  
    alunos/  
        ...  
    alunos.txt  
    disciplinas.txt  
    read-me.txt  
    main.c  
    alunos_funcoes.c  
    disciplinas_funcoes.c  
    editar_aluno.c  
    editar_aluno_funcoes.c  
    header.h  
    header_windows.h  
    header_osx.h
```

Arquivos:

alunos.txt: Arquivo onde é salvo em ordem alfabética o nome dos alunos com suas respectivas matrículas.

disciplinas.txt: Arquivo onde é salvo em ordem alfabética o nome das disciplinas com seus respectivos códigos e créditos.

read-me.txt: Informações básicas e exemplo de como testar a funcionalidade do programa.

main.c: Arquivo principal do programa, é onde está a função main() que dará início à execução do programa.

alunos_funcoes.c: Arquivo com as definições do Tipo Abstrato de Dados referente aos alunos.

disciplinas_funcoes.c: Arquivo com as definições do Tipo Abstrato de Dados referente às disciplinas.

editar_aluno.c: Arquivo com uma função principal chamada pela função main(), é o programa principal que permite a edição de alunos.

editar_aluno_funcoes.c: Arquivo com as definições do Tipo Abstrato de Dados referente à edição dos alunos.

header.h: Arquivo com todos os protótipos de funções e que faz a ligação entre os arquivos .c

header_windows.h e header_osx.h: Arquivos com a definição de códigos de pausa e de limpar a tela para cada sistema operacional.

Descrição das Estruturas de Dados Escolhidas:

Estrutura de dados tipo ALUNO:

```
struct aluno
{
    char * nome;
    char * matricula;
};

struct nodo_aluno
{
    struct nodo_aluno * anterior;
    struct aluno info;
    struct nodo_aluno * proximo;
};
```

A estrutura *struct aluno* contém o nome e a matrícula do aluno em questão.

O Tipo Abstrato de Dado escolhido para o tipo ALUNO foi a Lista Duplamente Encadeada, pois a possibilidade de poder avançar e voltar como desejar com a lista é de grande utilidade na hora de fazer a inserção ordenada que é pedida no problema.

Estrutura de dados tipo DISCIPLINA:

```
struct disciplina
{
    char * nome;
    char * codigo;
    int creditos;
};

struct nodo_disciplina
{
    struct nodo_disciplina * anterior;
    struct disciplina info;
    struct nodo_disciplina * proximo;
};
```

A estrutura *struct disciplina* contém o nome da disciplina, o código dela e seus créditos.

O Tipo Abstrato de Dado escolhido para o tipo DISCIPLINA foi a Lista Duplamente Encadeada, pois, como no dado do tipo ALUNO, é necessário fazer a inserção ordenada através do nome da disciplina, portanto a escolha da estrutura é similar.

Estrutura de dados tipo EDICAO:

```
struct editar
{
    int estado;
    char * nome;
    char * codigo;
    int nota;
    int vezes_cursadas;
};

struct nodo_editar
{
    struct editar info;
    struct nodo_editar * proximo;
};
```

O Problema consta que é necessário criar um modo de edição de aluno para poder inserir uma lista de disciplinas a ser cursadas, inserir notas, encerrar disciplinas e guardar as disciplinas encerradas com a constatação de que o aluno foi aprovado ou não.

Porém a implementação foi deixada livre, nesse caso, a estrutura de dados *EDICAO* é uma Lista Simplesmente Encadeada, por não conter nenhuma exigência específica e encapsula a estrutura *editar* que contém um estado, o nome de uma disciplina, o código de uma disciplina, a nota da disciplina e quantas vezes a disciplina foi cursada.

Esse tipo de dados cria um arquivo para cada aluno cadastrado no sistema, onde o nome do arquivo é o seu código de matrícula, nesse arquivo consta o Nome e a Matrícula do aluno e, além disso, uma lista com as disciplinas que estão sendo e que já foram cursadas.

Dentro do arquivo, isso é organizado da seguinte maneira:
[estado],[nome],[codigo],[nota],[vezes_cursadas]

Com essa estrutura, se torna possível manter o registro das disciplinas da vida acadêmica do estudante.

A variável **estado** indica se o aluno está cursando a matéria(0), se o aluno já foi aprovado na matéria (1) ou se o aluno já cursou a matéria e foi reprovado (2).

Caso o aluno tenha sido reprovado e a matéria seja colocada novamente na lista de disciplinas em andamento, a antiga, com a reprovação, não será apagada, será apenas adicionada à lista a mesma disciplina, com um novo estado e com as vezes que a disciplina já foi cursada.

Descrição da Usabilidade do Programa:

O Objetivo do programa é implementar a solução para a necessidade de um sistema de cadastro de alunos, disciplinas e de edição desses alunos, podendo fazer o controle das disciplinas que os alunos cursam.

O Arquivo Principal fornece 10 opções de interações divididas em Operações com Alunos e Operações com Disciplinas:

Operações com Alunos:

- 1 - Cadastrar Aluno
- 2 - Editar Aluno
- 3 - Remover Aluno
- 4 - Imprimir todos os alunos
- 5 - Gravar em arquivo

Operações com Disciplinas:

- 6 - Cadastrar Disciplina
- 7 - Remover Disciplina
- 8 - Imprimir todas as disciplinas
- 9 - Gravar em Arquivo

10 - Sair do Programa.

Opção 1: Cadastrar Aluno - ao escolher essa opção, será pedido o nome do aluno e seu número de matrícula, ele será inserido na lista de forma ordenada, será gerado um arquivo cujo nome é seu numero de matricula e que no interior tem seu nome e seu número de matricula, e o aluno será salvo no arquivo.

Opção 3: Remover Aluno - ao escolher essa opção, será pedido o código de matricula do aluno que deve ser removido, após isso, o usuário com a matricula especificada será removido da lista - mas o arquivo alunos.txt ainda não estará atualizado - caso a matricula não exista, nada é feito.

Opção 4: Imprimir todos os alunos: nessa opção, todos os alunos cadastrados na lista são mostrados na tela.

Opção 5: Gravar em Arquivo: essa opção é referente aos alunos e grava a lista de alunos no arquivo "alunos.txt"

Opção 6 : Cadastrar Disciplina - ao escolher essa opção, será pedido o nome da disciplina, o código dela e seus créditos, em seguida, ela será inserida na lista de forma ordenada, o arquivo "disciplinas.txt" não é alterado logo após a inserção

Opção 7 : Remover Disciplina - ao escolher essa opção, será pedido o código da disciplina, assim que especificado, o programa remove a disciplina com o código, se não existir uma disciplina com tal código, nada é feito.

Opção 8 : Imprimir todas as disciplinas - essa função mostra todas as disciplinas da lista na tela.

Opção 9 : Gravar em Arquivo - essa opção é referente às disciplinas cadastradas e grava essas disciplinas no arquivo "disciplinas.txt"

Opção 10 : Sair do Programa - termina a execução do programa.

Além desse menu de fácil uso, ao escolher a opção 2, Edição de Aluno, outro menu é aberto após digitar o número de matrícula do aluno que será editado.

Ao entrar nesse menu, o programa estará guardando o nome do aluno e seu número de matrícula.

Nesse outro menu, existem 7 opções referentes à edição de aluno:

- 1 - Inserir Disciplina sendo Cursada
- 2 - Ver Disciplinas em que o aluno foi aprovado
- 3 - Ver Disciplinas em que o aluno foi reprovado
- 4 - Ver Disciplinas que o aluno está cursando
- 5 - Inserir nota para uma disciplina em andamento
- 6 - Encerrar uma Disciplina
- 7 - Encerrar Edição

Opção 1: Inserir Disciplina sendo Cursada - nessa opção, deve-se informar o código da disciplina a ser inserida, então é verificado se a disciplina realmente existe, em seguida, se o aluno já não foi aprovado ou se não cursa a disciplina, depois é verificado se o aluno está cursando a disciplina pela primeira vez, se não estiver, a disciplina é inserida novamente, desde que ele não tenha sido reprovado 4 vezes, com o número de vezes cursadas, se o aluno ainda não tiver cursado a disciplina, ela é inserida normalmente.

Opção 2: Ver Disciplinas em que o aluno foi aprovado - Mostra na tela todas as disciplinas em que o aluno foi aprovado.

Opção 3: Ver Disciplinas em que o aluno foi reprovado - Mostra na tela todas as disciplinas em que o aluno foi reprovado.

Opção 4: Ver Disciplinas que o aluno está cursando - Mostra na tela todas as disciplinas que o aluno está cursando.

Opção 5: Inserir nota para uma disciplina em andamento - ao escolher essa opção, deve-se entrar com o código da disciplina e a nota a ser inserida, ela então é atualizada. Se o código não existir, nada é feito.

Opção 6: Encerrar uma disciplina - ao escolher essa opção, deve-se entrar com o código da disciplina a ser encerrada, se a disciplina estiver sendo cursada, ela é encerrada, caso o aluno tenha nota maior que 60, a disciplina muda o estado para aprovada, caso contrário, para reprovada.

Opção 7: Encerrar Edição - encerra o modo de edição e volta para o programa principal.

Outras Funcionalidades: Quando o programa é iniciado é criado duas listas, uma de aluno e uma disciplina com o conteúdo dos arquivos “alunos.txt” e “disciplinas.txt”

Quando o programa é encerrado, os arquivos “alunos.txt” e “disciplinas.txt” são automaticamente atualizados.

Atividades Realizadas por cada um do grupo:

Guilherme Pacheco de Oliveira - Projeto, Código e Relatório.