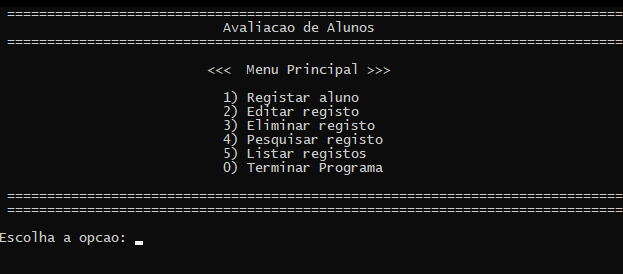
Algoritmos e Estruturas de Dados – Trabalho prático

Algoritmos e Estruturas de Dados

Programação em Web, Dispositivos e Aplicações Móveis

Carlos Xavier (57293), Fernando Marta (57437), Henrique Alves (57347), João Espada (57710), Telmo Reis (57346)



Índice

[Introdução 3](#_Toc44285515)

[Funcionamento do programa e opções tomadas 4](#_Toc44285516)

[Ecrã inicial 4](#_Toc44285517)

[1) Registar aluno 4](#_Toc44285518)

[2) Editar registo 5](#_Toc44285519)

[3) Eliminar registo 5](#_Toc44285520)

[4) Pesquisar registo 5](#_Toc44285521)

[5) Listar registos 6](#_Toc44285522)

[0) Terminar programa 6](#_Toc44285523)

[Conclusão 7](#_Toc44285524)

# Introdução

Neste relatório iremos retratar o funcionamento do programa e explicar as principais opções tomadas na construção do programa, conseguinte do trabalho prático de grupo da cadeira de Algoritmos e Estruturas de Dados proposto pelo professor Carlos Oliveira, que consiste em registar os alunos de uma turma e as respetivas avaliações na frequência e trabalho prático.

O programa deverá permitir ao utilizador diversas operações, nomeadamente: registar, editar, eliminar, pesquisar e listar por ordem crescente.

# Funcionamento do programa e opções tomadas

Na definição da estrutura do nosso programa decidimos criar dois ficheiros: um onde tem o menu principal (onde corremos o programa), e outro que trata de guardar todas as funções associadas ao funcionamento do programa.

A estrutura de dados adotada foi a de Lista Ligada Simples, pois achámos uma maior similitude com os registos de alunos que teríamos de manipular.

## Ecrã inicial

Antes de serem iniciadas as opções do “Ecrã inicial”, é invocada a função “importar()” que vai permitir ler e importar os dados presentes no disco, e carregá-los para memória, de modo a que seja possível interagir com os mesmos durante a sessão.

O ecrã inicial apresenta todas as opções a que o utilizador pode aceder para gerir os registos dos alunos da turma.

Estas opções são:

1. Registar aluno
2. Editar registo
3. Eliminar registo
4. Pesquisar registo
5. Listar registos
6. Terminar programa

### 1) Registar aluno

Esta opção serve para registar um novo aluno na base de dados.

Começa por pedir ao utilizador os elementos do registo de cada aluno: número, nome, nota da frequência e do trabalho.

Neste registo fazemos a validação do valor das notas, não permitindo notas inferiores a 0 ou superiores a 20.

A gravação do novo registo é inserida no início da lista, uma vez que é o método mais eficiente numa lista ligada para a inserção.

### 2) Editar registo

Esta opção serve para editar um registo existente na base de dados, de um aluno.

Nesta opção é chamada a função “perguntar()” que pede para inserir um número de aluno e guarda o mesmo temporariamente, servindo de input para a função “pesquisar()”. Esta função vai procurar e listar as informações do registo correspondente a um mesmo número de aluno

Após mostrar os dados do registo, é pedido ao utilizador que decida se pretende editar o mesmo ou não; se não pretender alterar nenhumas informações, o mesmo é redirecionado para o menu. Caso o utilizador decida que pretende editar o ficheiro selecionado, é então pedido ao utilizador que, campo a campo, digite as novas informações do registo previamente inserido.

De seguida é solicitado ao utilizado que selecione se deseja guardar as alterações feitas, ou se pretende descartá-las.

### 3) Eliminar registo

Esta opção serve para eliminar um registo existente na base de dados, de um aluno.

Tal como na opção “Editar registo”, na opção “Eliminar registo”, também é chamada a função “perguntar()” e “pesquisar()”, que têm a mesma utilidade da 2ª opção do Menu principal.

Após mostrar os dados do registo, é pedido ao utilizador que decida se pretende eliminar o mesmo ou não; se não pretender eliminar o registo, o mesmo é redirecionado para o menu. Caso o utilizador decida que pretende eliminar o ficheiro selecionado, o mesmo é, então, eliminado da lista.

### 4) Pesquisar registo

Esta opção serve para pesquisar um registo existente na base de dados, de um aluno.

Esta opção também chama as funções “perguntar()” e “pesquisar()”, que têm a utilidade referida anteriormente.

Além das duas funções acima mencionadas, é ainda utilizada uma terceira função, “espera()”, que tem como função aguardar o input na tecla “Enter”, que impede que o programa avance sem a mesma tecla ser pressionada, para que o utilizador possa tomar o tempo necessário para observar o registo pesquisado.

### 5) Listar registos

A 5ª opção serve para listar os registos existentes na base de dados de forma ordenada, em formato de tabela.

Nesta opção é chamada primeiramente a função “bubbleSort()” que tem como funcionalidade ordenar a lista, por número de aluno de forma crescente. Optámos por este algoritmo de ordenação, pois achámos que seria o algoritmo mais simples e fácil de implementar neste projeto. Apesar de não ser o algoritmo mais eficiente, torna-se irrelevante no nosso projeto dada a dimensão dos nossos dados.

Como forma de trocar os dados entre dois nós, utilizámos a função “trocar()” que invocamos dentro da função “bubbleSort()”.

De seguida, é chamada a própria função “listar()” que permite listar todos os registos inseridos da base de dados.

A seguir às funções “bubbleSort()” e “listar()”, é invocada a função “espera()”, que tem a mesma função utilizada na opção “Pesquisar registos”, para que o utilizador possa decidir quando deseja deixar de ver os registos listados.

### 0) Terminar programa

A 6ª e última opção, serve para que se possa terminar o programa, e, concorrentemente, salvar a informação em memória, no disco. Para tal, usamos a função “exportar()” que exporta e grava os dados da sessão em memória para o ficheiro “BDalunosAED.dat”, em disco.

# Conclusão

Ao concluirmos o programa,