Projeto



Elemento do grupo:

Luís Carlos Gonçalves Freitas,

Nº: 2029715,

Curso: Engenharia Informática.

Rodrigo Martim da Silva Gaspar,

Nº: 2028715

Curso: Engenharia Informática.

Carlos Miguel Alves Olim,

Nº: 2014015,

Curso: Engenharia Informática.

Carlos Diogo Félix Martins,

Nº: 2032215,

Curso: Engenharia Informática.

Estratégias e decisões tomadas

Nesta segunda fase do projeto, optamos por utilizar listas ligadas (uma para os alunos, outra para as refeições e por fim outra para a função de listar por ordem alfabética), para armazenar os dados referentes às diversas funcionalidades, devido ao facto de temos de utilizar alocação dinâmica de memória para esta última entrega.

Decidimos, ao inserir o aluno, inserir na lista ligada já por ordem do número, para tornar mais eficiente a pesquisa e a escrita para o ficheiro de texto, visto que era pedido para os alunos estarem ordenados pelo número de aluno no ficheiro. A mesma coisa foi feita para as refeições (os dados são introduzidos por ordem da data na lista).

Optamos também por criar uma função para converter todos os nomes dos alunos inseridos para letras maiúsculas, dado que depois na função de pesquisar pelo nome, o facto das letras serem minúsculas ou maiúsculas influência na pesquisa.

Na primeira entrega tinha-nos sido apontado como problema o facto de o programa falhar, por exemplo quando era esperado um inteiro e introduzíamos uma letra, não nos tínhamos preocupado com as validações. Nesta segunda entrega, tivemos esse cuidado em particular validamos todas estas situações, através da função que criamos “inputValidation” para evitar que o programa falhe por uma simples distração. De resto mantivemos mais ou menos a mesma ordem de ideias, visto as validações terem sido a nossa maior falha na primeira entrega.

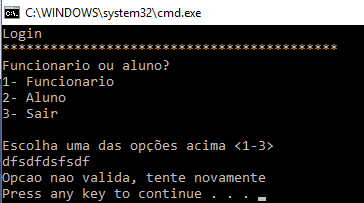
Optamos também por criar um código que interaja com o utilizador, com o objetivo de este ter sempre a consciência do que está a ser feito, visto acharmos ser um aspeto bastante importante também.

Corrigimos também o pormenor de nas funções de encomendar Refeições e de carregar plafond não ser preciso voltar a pedir o número de aluno, vamos buscar o número introduzido no login.

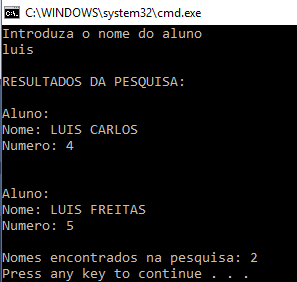
Fizemos a separação das funções a que o funcionário e o aluno podem aceder, tendo sido para o aluno (carregar plafond e encomendar refeições) e para o funcionário as restantes funções. Achamos que é o que faz mais sentido para este tipo de programa.

Exemplos de utilização da solução

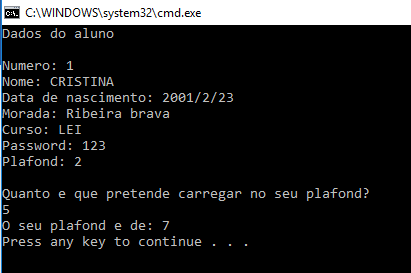
Exemplo de uma validação:



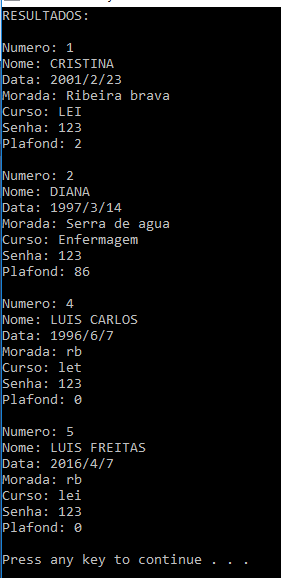
Exemplo da função de pesquisa pelo primeiro nome:



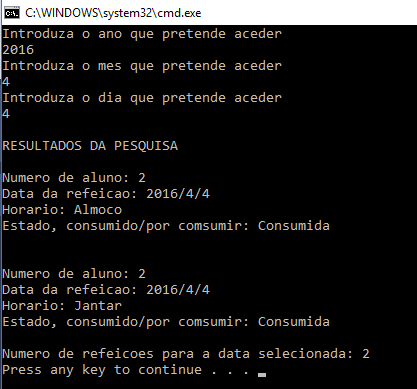
Exemplo da função de carregar o plafond:



Exemplo da função de ordenar por ordem alfabética:



Exemplo da função de listar refeições num determinado dia:



Comentários e conclusões

Concluindo, foi-nos bastante útil a realização deste trabalho, era um trabalho com várias funções que tínhamos de criar, funções de pesquisa, de listagem etc… O que nos levou a adquirir um conhecimento alargado da linguagem c++, para além daquele adquirido nas aulas, foi sem dúvida uma grande mais-valia este trabalho.

Encontramos algumas dificuldades ao longo da realização do trabalho, como por exemplo na parte de recarregar a informação do ficheiro de texto para as listas ligadas quando iniciamos o programa, as validações, etc, que fomos resolvendo ao longo do tempo através de pesquisas.

Foi muito interessante realizar este projeto, desenvolvemos um trabalho muito melhor do que o da primeira entrega, a primeira entrega serviu também para serem apontados alguns problemas que, que nesta segunda entrega já foram resolvidos.

O mais importante a reter é que nada é impossível na programação, podemos com alguma paciência desenvolver coisas espetaculares e que também podemos sempre melhorar o nosso código.

Ficamos também com a consciência de que as validações são fundamentais na realização de um programa, devemos sempre validar tudo o que é introduzido pelo utilizador.

Bibliografia

<http://www.cplusplus.com/doc/tutorial/>