



**POLITÉCNICO
DE LEIRIA**

ESCOLA SUPERIOR
DE TECNOLOGIA
E GESTÃO

RELATÓRIO PROJETO

PROGRAMAÇÃO I

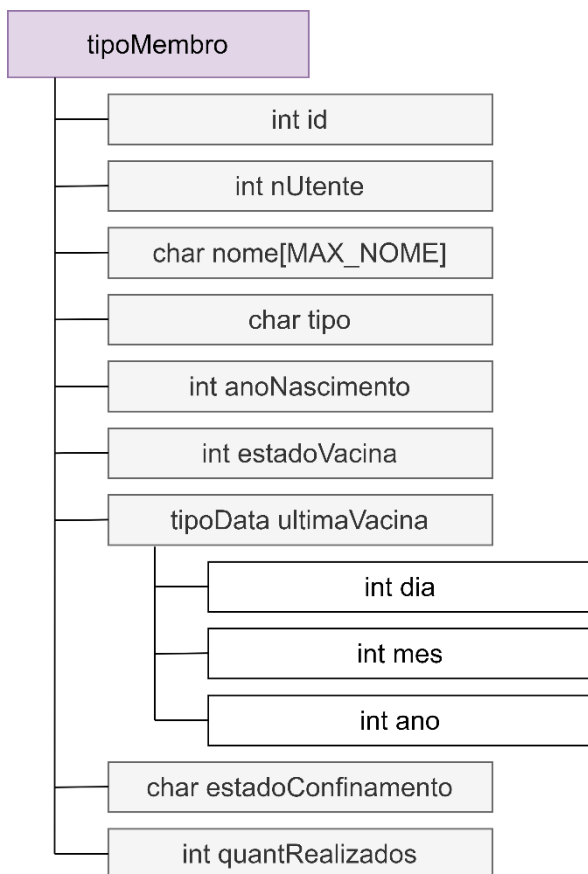
Trabalho realizado por:

2211444

Miguel Pedrosa Alberto

Estruturas

Para o vetor estático "vetorMembros", que armazena os dados de cada membro da comunidade usou-se a estrutura "tipoMembro":



Em que, **id** servirá para dar um Número de Identificação a cada membro registado, e coincidirá com a posição de cada membro no vetor. É utilizado recorrentemente no código do programa para facilitar algumas operações;

nUtente será o Número de Utente do membro;

nome será o nome do membro;

tipo será o tipo de membro. 'E' para Estudante, 'D' para Docente, e 'T' para Técnico.

anoNascimento será o ano de nascimento do membro;

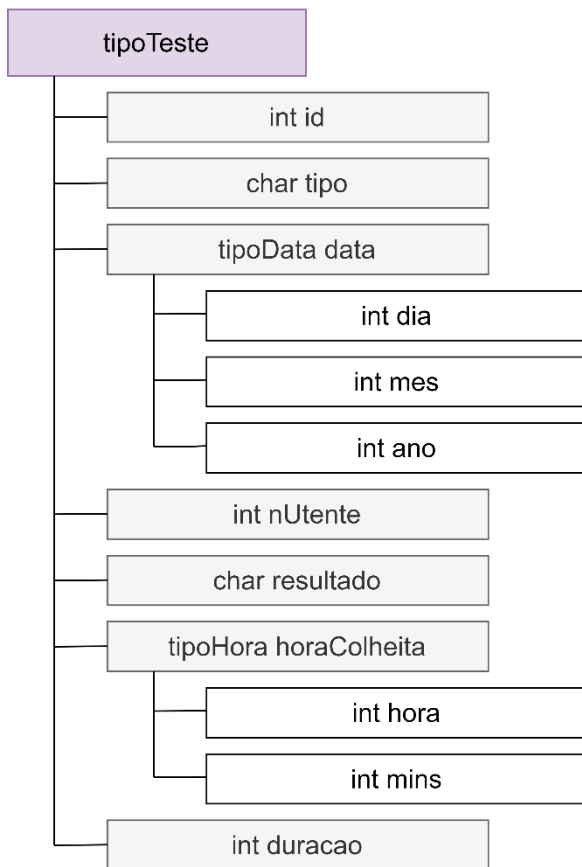
estadoVacina será o número de doses de vacina que o membro recebeu;

ultimaVacina será a data da última vacina que o membro recebeu;

estadoConfinamento será o estado de confinamento do membro. 'N' para membro "Não Confiando", 'Q' para membro "Em Quarentena", e 'I' para membro "Em Isolamento Profilático";

quantRealizados será o número de testes com resultado registado do respetivo membro. É utilizado para as estatísticas (ver membros com menos testes realizados).

Para o vetor dinâmico “vetorTestes”, que armazena os dados de cada teste agendado (e realizado, se for o caso) usou-se a estrutura “tipoTeste”:



Em que, **id** servirá para dar um Número de Identificação a cada teste registado (e “designação única”), e coincidirá com a posição de cada teste no vetor;

tipo será o tipo de teste. ‘P’ para teste PCR, e ‘A’ para teste Antigénico;

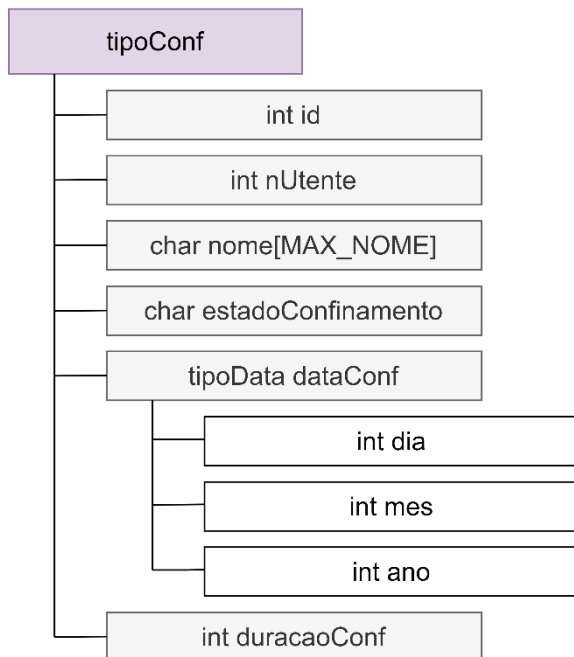
nUtente será o Número de Utente do membro que agendou o respetivo teste;

resultado será o resultado registado (aquando do seu registo pelo utilizador) do respetivo teste. ‘P’ para resultado “Positivo”, ‘N’ para resultado “Negativo”, e ‘I’ para resultado “Inconclusivo”;

horaColheita será a hora de realização do respetivo teste (aquando do registo do resultado pelo utilizador);

duracao será a duração, em minutos, que levou a realização do teste (aquando do seu registo pelo utilizador).

Para o vetor dinâmico “vetorConf”, que é usado para ler os dados de cada confinamento registado e listá-los no ecrã, tal como a variável “conf” que é usada para gravar e registar estes dados, usou-se a estrutura “tipoConf”:



Em que **id** servirá para dar um Número de Identificação a cada confinamento registado, e coincidirá com a posição de cada confinamento no vetor;

nUtente será o Número de Utente do membro que foi confinado;

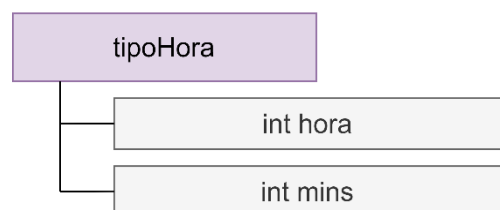
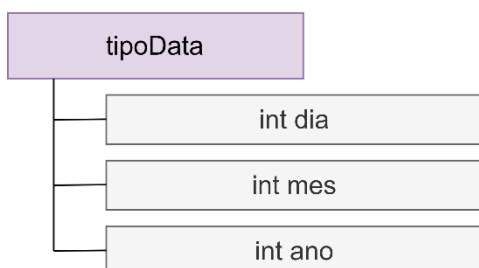
nome será o nome do membro que foi confinado;

estadoConfinamento será o estado de confinamento do membro confinado. ‘Q’ para membro “Em Quarentena”, e ‘I’ para membro “Em Isolamento Profilático”.

dataConf será a data em que o utilizador foi confinado (tanto pelo registo manual do utilizador como pelo registo automático devido a algum resultado comprometedor por parte de um teste);

duracaoConf será a duração, em dias, que o membro terá de ficar confinado. É registado pelo utilizador.

Estruturas utilizadas como “subestruturas” para as estruturas anteriores foram a estrutura “tipoData” e a estrutura “tipoHora”, que armazenam uma data e uma hora, respetivamente:



Organização Menu

O menu para o utilizador foi dividido em 5 submenus: Membros, Testes, Vacinações, Confinamentos e Estatísticas.

De forma geral, como os nomes indicam, o menu Membros trata das funções relativas aos dados dos membros da comunidade.

O menu Testes trata dos dados e das funções relativas aos testes de cada membro da comunidade.

De forma a facilitar a utilização do programa pelo utilizador; para a criação de submenus mais idênticos uns com os outros (a maioria dos menus tem uma parte para registo de dados seguido de uma parte para listagem de dados); para separar os dados dos membros com valores permanentes dos dados com valores que podem ser alterados; e para facilitar (eventualmente) a escalabilidade do programa, optou-se pela criação dos submenus de Vacinações e de Confinamentos (em vez de serem incluídos dentro do submenu de Membros). Estes permitem, respetivamente, o registo, pelo utilizador, de vacinações e de confinamentos de cada membro da comunidade. Tal como as respetivas listagens.

No menu Estatísticas, é possível ver dados estatísticos extras, tal como a listagem dos membros com menos testes registados e a listagem dos testes mais recentes.

Para além destes 5 submenus referidos, o utilizador tem, no menu principal, a opção de carregar manualmente os dados anteriormente gravados (através da leitura dos ficheiros binários), e a opção de gravar manualmente os dados (através do seu armazenamento nos ficheiros binários). De referir, que o utilizador não precisa de realizar estas opções, uma vez que os dados são carregados logo no início do programa e são gravados após a introdução de dados novos.

O utilizador tem ainda a opção para sair do programa.

Notas

- Em vez de usar uma variável diferente para cada vetor como contador, utilizou-se um vetor que guarda todos os contadores e/ou quantidades, chamado de “vetorQuant”. Este vetor, com tamanho 5 guarda, respetivamente, a seguinte informação:

```
vetorQuant[0] == Quantidade de Membros  
vetorQuant[1] == Quantidade de Membros Vacinados  
vetorQuant[2] == Quantidade de Testes Agendados  
vetorQuant[3] == Quantidade de Testes com Resultado Registrado  
vetorQuant[4] == Quantidade de Confinamentos Registrados
```

- Algumas das funções poderiam ter sido resolvidas de forma a chamar outras funções (também usadas noutras funções *callers* diferentes) para encurtar a totalidade de linhas de código. E em muitas funções, isso é feito. No entanto, para outras, para simular uma melhor experiência para o utilizador, como mensagens personalizadas para cada função/situação específica (em vez de predominarem mensagens “generalistas”), houveram alguns casos em que se optou por repetir algumas linhas de código, com *printf*s com mensagens diferentes/personalizadas (dependendo da situação).

- Ao executar o programa fez-se a leitura dos ficheiros anteriormente gravados “automática”, com a leitura dos ficheiros que contêm a informação dos vetores, logo depois das declarações e inicializações das variáveis do *main*. De igual forma, fez-se o armazenamento dos dados automática (depois da inserção de novos dados ou alterações de dados), nos respetivos ficheiros.

Apesar disto, deixou-se a opção de ler e a opção de armazenar os dados manualmente, pelo utilizador, no menu principal, como referido anteriormente.

- Considerou-se que cada membro pode ter vários testes agendados, desde que não sejam do mesmo tipo e à mesma data.

- No registo de um novo membro da comunidade, o seu estado de vacinação toma o valor predefinido de 0 doses e o seu estado de confinamento toma o valor predefinido de “Não Confinado”. Para haver alterações a estes valores, por parte do utilizador, este tem de utilizar os menus próprios para o efeito (menu “Vacinações” e menu “Confinamentos”).

- No registo do resultado de um teste de diagnóstico foram feitas duas funções. Uma que pede o ID do respetivo teste, e outra que pede o Número de Utente do Membro do respetivo teste. Com esta última será apresentada uma lista com os testes desse membro para o utilizador poder escolher qual a registar. Caso só haja um teste por registar, este será automaticamente escolhido. É pedido, de seguida, os dados ao utilizador.

Cada uma destas funções tem uma opção diferente (para ser escolhida pelo utilizador) no menu de “Testes>Registar Teste”.

- Na função de atualizar o estado de vacinação de membro, optou-se por ser possível atualizar o respetivo estado do membro para um número de doses igual ou menor ao já estabelecido, dando um aviso prévio ao utilizador (perguntando se pretende continuar ou não). Isto serviria, por exemplo, para corrigir a data de vacinação do membro, e/ou o seu número de doses, para corrigir algum engano por parte do utilizador, durante a inserção de dados anterior. Se o utilizador inserir um número de doses superior ao já estabelecido não aparecerá qualquer aviso.

- Relativamente aos confinamentos e listagens de confinamentos ocorridos optou-se por uma abordagem diferente da usada para os Testes. Enquanto que nos Testes o próprio vetor (vetorTestes) vai sendo atualizado com os dados recebidos, para os Confinamentos optou-se por usar ficheiros para guardar a informação relativa aos Confinamentos. Assim, sempre que um confinamento é registado, este é guardado no ficheiro “logConfinamentos.dat”, e para a listagem dos Confinamentos é feita a leitura deste ficheiro.

- Logo após o registo do resultado de um teste, poderá ser necessário o utilizador agendar um novo teste PCR (para cumprir a proposta do enunciado). De igual forma, (e também para cumprir a proposta do enunciado) será registado um novo confinamento “automaticamente”, partindo da data de agendamento do próprio teste. Caso isto aconteça faltará apenas registar a duração desse confinamento, que será pedido ao utilizador para preencher.

- Devido à forma como se optou por listar os dados é aconselhável estender a largura da janela de execução do programa, se necessário, para uma melhor visualização do programa.

