

1. Objetivo

Pretende-se desenvolver um programa que auxilie na gestão dos testes de diagnóstico da covid-19 e dos confinamentos dos membros da comunidade académica afetos a um determinado curso. Considera-se que essa comunidade terá no máximo 200 elementos.

2. Descrição do problema

O programa deverá registar a informação referente aos membros da comunidade académica, aos testes de diagnóstico agendados e aos respetivos resultados.

Os testes só poderão ser agendados para membros da comunidade académica que se encontrem registados no programa. Ao efetuar o agendamento deverá ser identificado o tipo de teste (PCR ou antigénio) a realizar.

Após a realização de um teste previamente agendado, o respetivo resultado deverá ser registado. Caso o resultado seja inconclusivo deverá ser agendado um teste do tipo PCR. O mesmo deverá suceder para os resultados positivos obtidos em testes do tipo antigénio.

Um membro da comunidade académica deverá ser colocado imediatamente em isolamento profilático quando obtiver um resultado positivo num teste. Caso o resultado do teste seja inconclusivo o membro da comunidade deverá ficar em quarentena.

Os membros da comunidade académica poderão também ficar em isolamento devido a contactos de proximidade com um infetado. Estes casos são registados pelo utilizador.

3. Descrição dos elementos

Membro da comunidade académica: Cada membro da comunidade académica é caracterizado por um número de utente de saúde (único), pelo nome, pelo tipo de membro (estudante, docente ou técnico), pelo ano de nascimento, pelo estado de confinamento (não confinado, quarentena ou isolamento profilático), pelo estado de vacinação (sem vacina, dose1, dose2 ou dose3) e pela data da última vacina. A comunidade possui, no máximo, 200 elementos.

Testes de diagnóstico: Cada teste é caracterizado por uma designação (única), pelo tipo de teste (PCR ou antigénio), pela data de realização e pelo número de utente de saúde do membro da comunidade académica.

Para cada teste será também necessário registar o resultado (positivo, negativo ou inconclusivo), a hora da colheita e o tempo de duração (em minutos).

4. Funcionalidades da aplicação a desenvolver

A aplicação, a desenvolver na linguagem C, deverá apresentar ao utilizador um menu de opções que disponibilize as seguintes funcionalidades:

- Apresentar no menu principal a seguinte informação: quantidade de membros da comunidade, quantidade de testes agendados, quantidade de testes realizados e quantidade de membros vacinados. ✓
- Inserir e listar a informação referente aos membros da comunidade. Deverão também ser listadas, para cada membro, a quantidade de testes realizados e a informação (tipo de teste, data) dos testes agendados. ✓
- Registrar/atualizar o estado de vacinação de um membro da comunidade. ✓
- Registrar/atualizar o estado de confinamento de um membro da comunidade. ✓
- Agendar a realização de um teste, considerando que num determinado dia se poderão efetuar, no máximo, 15 testes do tipo PCR. ✓
- Listar os dados de todos os testes. Para cada teste realizado deverão também ser mostrados o resultado, a hora de realização e a duração. ✓
- Alterar a data agendada para a realização de um teste. ✓
- Registrar o resultado de um teste previamente agendado. Toda a informação do teste, acrescida do nome e do estado da vacinação do membro da comunidade que o realizou, deverá ser acrescentada num log, em ficheiro de texto. ✓
- Armazenar (e ler) em ficheiro(s) binário(s) a informação referente aos membros da comunidade académica e aos testes agendados. ✓
- Apresentar a informação de um teste (selecionado pelo utilizador), incluindo a seguinte informação do respetivo membro da comunidade académica: nome, tipo de membro e quantidade de testes positivos. ✓
- Apresentar os seguintes dados estatísticos: quantidade de cada tipo de membro da comunidade académica, tempo médio de duração de cada teste realizado, percentagem de testes inconclusivos, membro(s) da comunidade académica com a menor quantidade de testes realizados, e teste(s) realizado(s) mais recente.
- Apresentar uma listagem dos casos de confinamento ocorridos contendo a seguinte informação: membro da comunidade académica, tipo de confinamento (quarentena ou isolamento profilático), data de confinamento e tempo de duração (em dias).

5. Notas Importantes

- Para assegurar uma boa implementação do Projeto deverá previamente identificar as principais estruturas de dados a utilizar e esboçar o algoritmo.
- O Projeto deverá estar dividido em várias funções e dividido por vários ficheiros, de forma a facilitar a sua correta estruturação, implementação e entendimento, e para evitar a repetição de código.
- Deverão ser efetuadas todas as validações e proteções necessárias ao correto funcionamento do programa.
- O programa deverá apresentar mensagens de aviso/erro sempre que necessário.
- Deverá ser dada prioridade à implementação das funcionalidades requeridas, em detrimento da implementação de funcionalidades adicionais.
- A funcionalidade do programa é mais relevante do que a apresentação da informação.

6. Regras e informações complementares

1. A **Fase 1** do Projeto deverá ser realizado em **grupos de dois estudantes** do mesmo turno PL.
2. Prazo para inscrição do grupo (no moodle): **18 dezembro 2021**.
3. Prazos para **submissão do projeto** (através do moodle): **19 janeiro 2022**.
4. **O projeto só será considerado definitivamente submetido após obterem essa informação do moodle.**
5. Na **Fase 1** deverão ser entregues os seguintes elementos em formato digital:
 - O **projeto** (ficheiros de código fonte), que terá de **compilar** no gcc utilizando o IDE *Codeblocks*.
 - **Relatório** contendo obrigatoriamente a descrição e representação gráfica das estruturas de dados utilizadas, e a identificação das funcionalidades não desenvolvidas ou parcialmente/incorrectamente desenvolvidas.
Poderá também conter a justificação de algumas opções de implementação tomadas.
6. A **Fase 2** do projeto é individual, realizando-se a **31 janeiro 2022**, e consiste na elaboração de código adicional à versão entregue na Fase 1.
7. Data prevista para afixação da nota do projeto: **19 fevereiro 2022** (nota final).
8. Poderão ser agendadas **defesas de projetos** quando surgirem dúvidas quanto à sua autenticidade.

7. Cotações

Nota do Projeto (0-20) = 50% * notaFase1 (mín de 8.0) + 50% notaFase2 (mín de 8.0)

Critérios de correção da Fase 1 do Projeto e respetivas cotações (*)

a) Relatório contendo todos os elementos obrigatórios	2,5%
b) Estrutura de Menus (incluindo informação solicitada)	5,0%
c) Gestão de membros da comunidade, e respetivas listagens	10,0%
d) Registar/atualizar o estado de vacinação	5,0%
e) Registar/atualizar o estado de confinamento.....	2,5%
f) Agendamento e listagem dos testes de diagnóstico.....	12,5%
g) Alteração de teste agendado	2,5%
h) Registo de resultado de um teste de diagnóstico (e armazenamento em <i>log</i>).....	12,5%
i) Ficheiros (leitura e escrita de dados dos membros da comunidade, e dos testes)	10,0%
j) Informação de um teste de diagnóstico	5,0%
k) Dados estatísticos (cálculo e apresentação)	15,0%
l) Listagem de casos de confinamento	12,5%
m) Estrutura geral do programa, incluindo a adequada utilização de variáveis locais, constantes, funções, comentários, indentação do código, separação do código por ficheiros	5,0%

(*) Com exceção da alínea a), **as cotações apenas serão aplicadas em projetos que compilem sem erros**, não se considerado o código comentado.

Apenas serão cotados os **relatórios contendo todos os elementos obrigatórios** (descrição e representação gráfica das estruturas de dados, e a identificação das funcionalidades não desenvolvidas ou parcialmente/incorrectamente desenvolvidas).

Será anulado qualquer projeto total ou parcialmente plagiado, podendo os estudantes envolvidos ser sujeitos a **procedimento disciplinar**.

É considerado plágio a cópia de textos, documentos, imagens ou código de outras fontes, e considerá-lo como próprio.

Essa situação pode ocorrer quando é partilhado código entre grupos de projeto, ou quando o projeto é desenvolvido por elementos de vários grupos.