

Algoritmos e Estrutura de Dados

Trabalho prático 1

Horários de estudantes L.EIC

Grupo 106:

- . Gonçalo Jorge Soares Ferreira - up202004761
- . Martim Raúl da Rocha Henriques - up202004421
- . Simão Queirós Rodrigues - up202005700



Problema

O trabalho prático tem como objetivo o desenvolvimento de um sistema capaz de ajudar na gestão de horários após a sua elaboração, tais como:

- Visualização de horários.
- Remoção de estudantes de uma turma/UC.
- Adição de estudantes a uma turma/UC.
- Alteração de turma/UC aos estudantes.



Solução

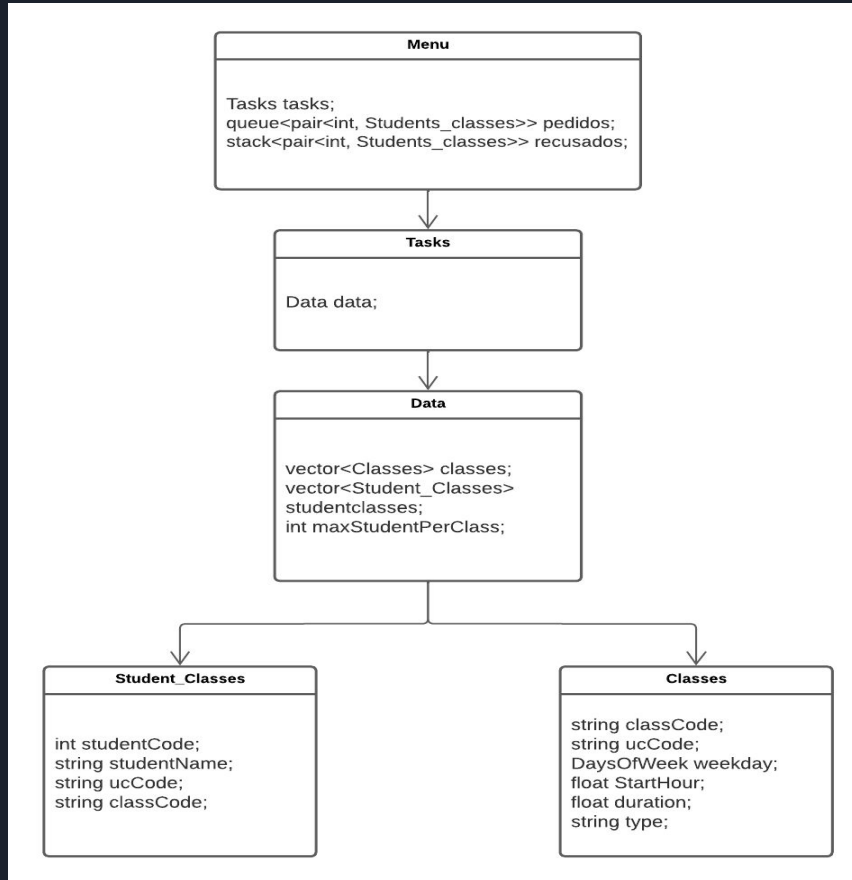
- Como se trata de um problema do mundo real, utilizamos a programação orientada a objetos.
- Criamos classes para definir as funções e atributos de cada entidade envolvida no problema.
- Uso de estruturas de dados lineares no contexto mais apropriado.
- A interação com o programa ocorre através do terminal, onde o usuário pode, através de comandos numéricos e inputs de texto, realizar as operações que quiser, caso sejam possíveis.



Algoritmos relevantes

- Algumas classes possuem listas como atributos, e o algoritmo mais usado pelo grupo para encontrar elementos nelas foi a busca linear, que tem como complexidade $O(n)$.
- Para as listagens em ordenações diferentes, foi usado o método `sort()` da STL em listas.

Diagrama de classes





Funcionalidades implementadas

Ao iniciar o programa, o usuário tem um menu à sua disposição com diferentes opções de escolha, sendo elas:

- Listagem.
- Remover um estudante de uma turma/UC (OK/Completa).
- Adicionar um estudante a uma turma/UC (Parcial).
- Alterar a turma/UC de um estudante (OK).
- Alterar um conjunto de turmas/UCs de um estudante (OK).



Funcionalidades implementadas

Ainda dentro da opção 1 do menu (listagem) é exibido outro menu com as seguintes opções, cujas funcionalidades estão OK/Completas:

- Pessoas por turma.
- Pessoas por ano.
- Pessoas por UC.
- Lista de estudantes.
- Lista de estudantes por turma.
- Lista de estudantes por ano.
- Lista de estudantes por UC.
- Horário de um estudante.
- Lista de estudantes com mais que um certo número de UCs.
- Voltar atrás.



Destaque de funcionalidade

Uma funcionalidade que merece destaque é a adição de um estudante a uma turma/UC.

Dado que exige o cumprimento de certas restrições, além de serem fundamentais também para a alteração de turma/UC de um estudante, bem como um conjunto delas.

Esta função começa por filtrar os dados para obter o horário do estudante e o horário da turma para que se quer adicionar. Com esta filtração conseguimos simplificar o problema ficando com um menor volume de dados para gerir, sendo bem mais fácil a conclusão de poder ou não adicionar o estudante à turma.



Dificuldades encontradas

- Utilização da BST.
- Adição de um estudante a uma turma/UC cumprindo as restrições.

Participação dos membros: os três membros trabalharam e colaboraram igualmente no projeto.