

Algoritmos e Estrutura de Dados

Sistema de Horários de estudantes L.EIC

Turma 14, Grupo 72

Anete Medina Pereira - up202008856

— Gabriel Anibal dos Santos Sousa - up202108816 —

Matheus Negrini Liotti - up202102363

Descrição do Problema

Desenvolver um sistema que auxilie na gestão de horários após sua elaboração.

O sistema tem como objetivo guardar informação relativa aos estudantes, respetivos horários e turmas.

É possível a troca de turma por parte do estudante, tendo em conta a ocupação da turma para a qual este deseja ir. É também permitida a adição ou remoção de um estudante.

O sistema dá também a possibilidade do utilizador visualizar a ocupação de determinada turma, unidade curricular ou ano.

Descrição da Solução

Para resolver o problema proposto procedemos à criação das classes **LogisticaDeTurmas**, **TrocaDeTurmas** e **ListingBasedOnOccupation** que têm como atributos listas e vetores de objetos das outras classes. Outras classes que considerámos fundamentais.

- **Estudante:** Guarda o nome, código do estudante e uma lista de turmas às quais o estudante pertence.
- **Aula:** Contém informações sobre os dias da semana, horário de início e fim, duração, códigos das UC's e códigos das aulas.
- **Turma:** Armazena informação sobre as UC's de determinada turma bem como os estudantes nela presentes.

Criámos também classes secundárias com **PedidoDeTrocaDeTurmas**, **Menu** e **ReadFiles** que também foram importantes para a construção das outras classes.

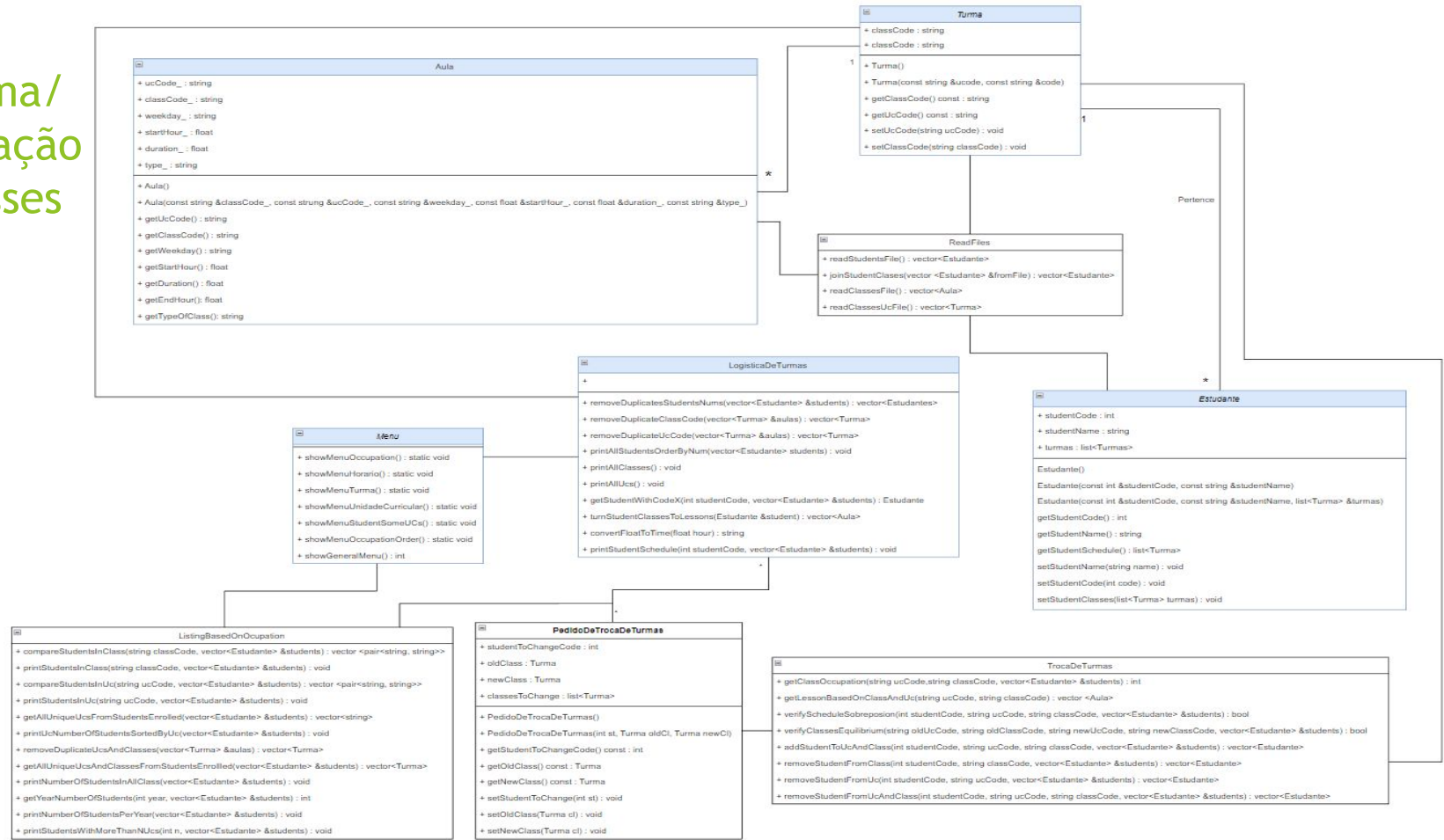
Identificação de Algoritmos Relevantes

Algoritmos de Ordenação: para realizar a ordenação do sistema usamos em sua maioria vetores e listas.

Algoritmos de Pesquisa:

- Linear Search: Utilizamos este método para pesquisar estruturas ordenadas e não ordenadas.

Diagrama/ Enumeração de Classes



https://drive.google.com/file/d/1yEbtkiLCMHWWFDI_UHzoQ228P0Rdc-z/view?usp=sharing

Funcionalidades Implementadas

- **Listagem:**

O utilizador tem a possibilidade de fazer a listagem dos alunos pertencentes a uma turma, unidade curricular ou ano, bem como a ocupação destes.

O utilizador tem também a possibilidade de fazer a listagem do horário de um determinado estudante.

- **Pesquisa:**

O utilizador consegue pesquisar pelos alunos que frequentam mais do que N unidades curriculares.

- **Funcionalidades extra:**

É também permitido ao utilizador adicionar ou remover um estudante de uma turma ou unidade curricular.

Funcionalidades Implementadas

- Leitura de dados e armazenamento em estruturas de dados lineares: **Completa**
- Ocupação de turmas/ano/Uc: **Completa**
- Horário de um determinado estudante: **Completa**
- Estudantes inscritos em determinada turma/Uc/ano: **Completa**
- Estudantes com mais de n Ucs: **Completa (ordenados alfabeticamente)**
- Remover estudante de turma/Uc: **Completa**
- Adicionar estudante a uma turma/Uc: **Completa**
- Alterar turma/Uc de um estudante: **Completa**

Funcionalidades Extras

- Listagem de todas as Ucs disponíveis no sistema
- Listagem de todas as turmas para cada uc disponíveis no sistema
- Remover estudante de uma uc simplesmente
- Remover estudante de uma turma simplesmente

Destaque de Funcionalidade

Das várias funcionalidades implementadas no trabalho aquela que consideramos ser a melhor é a de adicionar um estudante a uma turma presente na função `addStudentToUcAndClass`.

Isto porque esta funcionalidade implicou a criação de várias funções auxiliares para verificar se um estudante pode ou não ser adicionado à nova turma, e implicou também o uso de outras funções de outras classes para imprimir o horário do estudante antes de ser adicionado à nova turma e depois de ser adicionado se houver mudanças no horário.

Dentre as funções auxiliares implementadas destacamos a `verifyScheduleSobreposion` que verifica a sobreposição do horário atual do estudante com o horário das novas aulas, e a `verifyClassesEquilibrium` que verifica o equilíbrio entre as aulas. Também é verificado se o número de estudantes da turma a que o estudante deseja ir não é superior a 30 que foi definido para nós como a capacidade máxima de uma turma.

Exemplos de execução

- Número de alunos inscritos em cada ano

```
escolheu: 2. Contagem de alunos
-----
Menu Alunos
1. Visualizar Contagem de alunos em todas as ucs
2. Visualizar Contagem de alunos em todas as turmas
3. Visualizar Contagem de alunos em todos os anos
4. Sair
Escolha uma das opcoes!3
===== Numero de alunos inscritos em cada ano =====
Nr de estudantes no 1o ano: 140
Nr de estudantes no 2o ano: 496
Nr de estudantes no 3o ano: 333
-----
```

Exemplos de execução

- Estudantes com mais de N Ucs (neste caso 6)

```
0. Sair
1. Listing de estudantes inscritos em pelo menos uma unidade curricular
2. Contagem de estudantes em cada Uc, Turma e Ano
3. Visualizar Horário de estudante de acordo com o seu numero
4. Visualizar estudantes Inscritos numa Turma
5. Visualizar estudantes inscritos numa Unidade Curricular
6. Visualizar estudantes com mais do que n Uc's
7. Listing de todas as Unidades Curriculares disponiveis no sistema
8. Listing de todas as turmas disponiveis no sistema
9. Operacoes para adicionar, remover e trocar de turma
Escolha uma das opcoes!
```

6

```
Escolheu:6. Visualizar estudantes com n Uc's
```

```
-----
```

```
Digite o numero de Unidades Curriculares:6
```

```
Amelia
Bento
Bianca
Bruna
Eunice
Filipa
Francisca
Joao Rodrigo
Jose Nelson
Leonor
Manuel Carlos
Marta
Nicole
Raul
Valentina
```

Exemplos de execução

- Remover estudante de uma turma/uc

```
Escolheu:8. Troca de turmas
-----
Menu Alunos
1. Remover estudante de uma uc
2. Remover estudante de uma turma
3. Remover estudante de uma uc/turma
4. Adicionar estudante a uma uc/turma
5. Trocar estudante de uc/turma
6. Sair
Escolha uma das opcoes!3
Digite o numero de estudante:202031607
Digite o codigo da uc:L.EIC004
Digite o codigo da turma:1LEIC08
=====
```

```
===== Turmas antigas =====
===== Horario do/a Gisela =====
Unidade Curricular: L.EIC004          Turma: 1LEIC08
Dia da semana: Thursday
Tipo de Aula: TP
Hora de Inicio: 08:30          Hora de fim: 10:30
Duracao: 2
-----
Unidade Curricular: L.EIC004          Turma: 1LEIC08
Dia da semana: Thursday
Tipo de Aula: T
Hora de Inicio: 11:00          Hora de fim: 13:00
Duracao: 2
-----
Unidade Curricular: L.EIC005          Turma: 1LEIC08
Dia da semana: Wednesday
Tipo de Aula: TP
Hora de Inicio: 10:30          Hora de fim: 12:30
Duracao: 2
-----
Unidade Curricular: L.EIC005          Turma: 1LEIC08
Dia da semana: Friday
Tipo de Aula: T
Hora de Inicio: 08:00          Hora de fim: 10:00
Duracao: 2
-----
```

```
-----
===== Turmas novas =====
===== Horario do/a Gisela =====
Unidade Curricular: L.EIC005          Turma: 1LEIC08
Dia da semana: Wednesday
Tipo de Aula: TP
Hora de Inicio: 10:30          Hora de fim: 12:30
Duracao: 2
-----
Unidade Curricular: L.EIC005          Turma: 1LEIC08
Dia da semana: Friday
Tipo de Aula: T
Hora de Inicio: 08:00          Hora de fim: 10:00
Duracao: 2
-----
```

Dificuldades Encontradas e esforço de cada elemento do grupo

Encontramos dificuldades na manipulação da estrutura de Árvores Binárias (houveram problemas na implementação e entendimento do código, portanto não foi utilizada).

Para gerir a documentação Doxygen houveram problemas para conversão no formato html e utilização de softwares auxiliares.

Os testes para correr algumas funcionalidades geraram confusão devido ao uso de tipos e variáveis equivocados.

Grande parte da implementação das funcionalidades foi feita pela Anete, o Gabriel e o Matheus fizeram mais coisas como o powerpoint, o menu, o diagrama UML, setters, getters e doxygen (todos se reuniram para revisão do trabalho).