```
Projeto 'Schedule' {
  [PEDRO CASTRO, LEONARDO GARCIA E MARCEL
  MEDEIROS]
    < TURMA 10 - EQUIPA 4 >
```





```
Indice {
    01
          DEMONSTRAÇÃO PRÁTICA
          < Listagem e exemplos de
          funcionalidades implementadas >
                02 CLASSES E METÓDOS
                      < Classes criadas para alocar as entidades
                      trabalhadas e como se relacionam >
                            03
                                 DATA STRUCTURES
                                  < Evidênciação das estruturas de
                                  dados >
```





### **ADMIN**

1

- Consulta UCs com mais estudantes e suas turmas
- Consulta estudantes (sorting por ano, UC e turma)
- Consulta estudantes inscritos em pelo o menos N Ucs
  - 11
- Consulta todos os requests
  - 13

14





### DEMONSTRAÇÃO PRÁTICA

```
Uc's with most students:

L.EIC013 - 400
L.EIC015 - 357
L.EIC014 - 352
L.EIC012 - 342
L.EIC011 - 329
L.EIC023 - 281
L.EIC025 - 273
L.EIC021 - 262
L.EIC022 - 223
L.EIC022 - 85
L.EIC003 - 74
L.EIC004 - 71
L.EIC005 - 50
L.EIC001 - 32
```

Enter the number of UC's:
-> 6

There are 112 students registered in at least 6 Uc's.

[0] Go back to the Administrator menu

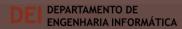
### **STUDENT**

1

- Consulta horário
- Consulta UCs inscritas e suas respectivas turmas.
- Consulta estudantes (sorting por ano, UC e turma)
- □ Faz requests
- Desfaz requests
- Consulta seus requests

14





### DEMONSTRAÇÃO PRÁTICA

->

```
Gabriela's schedule:
UC: L.EIC011 | Class: 2LEIC12 | Day: Tuesday | Begin:
                                                     8 Ends:
UC: L.EIC015 Class: 2LEIC10 Day: Tuesday Begin: 10.5 Ends: 12.5 Type: TP
UC: L.EIC011 Class: 2LEIC12 Day: Wednesday Begin: 8.5 Ends: 10.5 Type: TP
UC: L.EIC005 Class: 1LEIC05 Day: Wednesday Begin: 10.5 Ends: 12.5 Type: TP
UC: L.EIC004 Class: 1LEIC08 Day: Thursday Begin: 8.5 Ends: 10.5 Type: TP
UC: L.EIC004 | Class: 1LEIC08 | Day: Thursday | Begin:
                                                  11 Ends: 13 Type: T
UC: L.EIC011 Class: 2LEIC12 Day: Thursday Begin: 14.5 Ends: 15.5 Type: T
UC: L.EIC015 Class: 2LEIC10 Day: Thursday Begin: 15.5 Ends: 17.5 Type: T
UC: L.EIC014 Class: 2LEIC14 Day: Thursday Begin: 17.5 Ends: 19.5 Type: PL
UC: L.EIC005 | Class: 1LEIC05 | Day:
                                  Friday | Begin:
UC: L.EIC012 Class: 2LEIC10 Day:
                                  Friday Begin:
                                                   14 Ends:
                                                              16 Type: TP
UC: L.EIC014 Class: 2LEIC14 Day:
                                  Friday Begin:
UC: L.EIC012 Class: 2LEIC10 Day:
                                  Friday | Begin:
                                                    18 Ends:
[0] Go back to the Student menu
                                                       Enter your option:
                                                        [1] Enroll in a new Uc
                                                        [2] Disenroll from a Uc
                                                        [3] Switch Uc
                                                        [4] Switch classes
                                                        [0] Go back to the Student menu
```

# CLASSES {





< Caracterizada por seu atributo UcCode e armazena suas turmas >



### ecture

< Representa uma aula. Suas instâncias são usadas para armazenar dados de classes.csv >



Request < Implementa todos os tipos de request e suas eventuais condicionais de verificação



### Student

< Atributos como código, nome e schedule são capazes de armazenar dados de students classes.csv >



< Não contém atributos. Criada para definição de metódos de acesso ao dataset e implementação de funcionalidades >



## App

< Utiliza de todas as classes mencionadas para criar a interface do projeto >





# Data Structures{

# ALGORITMOS E ESTRUTURAS DE DADOS

## List

O container permite inserção e remoção em **complexidade O(1),** assim, aproveitamos desta vantagem para implementar métodos como **loadLecture** e **classesCheck** 

# Vector

Utilizamos de sua alocação dinâmica e capacidade de iterar por ponteiros e iteradores em atributos como **UcClasses** em Uc e em métodos como **studentInUc** em Script.

# Queue

Sua grande vantagem é o contexto operacional **FIFO.** Utilizamos disto para implementar containers como **eligibleClasses** na função classesCheck (removação do topo O(1)) e para **reescrever datasets.** 

### Stack

Seu contexto **LIFO** nos permitiu fazer operações de sorting como **printClasses** em ordem decrescente.



### Set

A grande vantagem do container é sua habilidade de ordenar seus elementos. Usamos disto para criar o horário do estudante, com um operator overload de '<' para adequar o horário em dias da semana e horas do dia. Além disto, outra vantagem é a restrição que garante que dois elementos não possuam o mesmo valor, essa vantagem foi usada em métodos como studentInYear.

```
Projeto 'Schedule' {
     Alguma Pergunta?
  Acesso em -> github.com/peucastro/aed_project
```



