

Programando em Java

(Ordenação e Iteração Ordenada de Vectores)

Mestrado Integrado em Engenharia Informática FCT UNL

<http://ctp.di.fct.unl.pt/miei/ip/>

Corpo Docente 2020/2021

António Ravara, Artur Miguel Dias, Bernardo Toninho,
Ema Vieira, Inês Fernandes, Margarida Mamede,
Miguel Monteiro, Rui Nóbrega

Laboratório (adaptado de teste 2017/18)

O laboratório **X** gere projectos de cientistas; cada projecto necessita de vários investigadores.

Cada investigador tem um *email* único, que o identifica, e pode propor projectos no laboratório.

Cada projecto é identificado por um número único e é caracterizado por uma descrição, pelo *email* do seu proponente (que é um investigador), pela quantidade de investigadores colaboradores necessários no projeto e pelos *emails* dos colaboradores efectivos (os colaboradores que existem em cada momento).

Nenhum investigador pode ser simultaneamente proponente e colaborador no mesmo projeto.



Laboratório de Investigação

É sempre possível adicionar projectos ao laboratório.

Também se podem associar colaboradores a projectos, sempre e quando ainda estejam em falta.

Para além disso, é sempre possível consultar os projectos de um dado investigador.



Laboratório de Investigação

Que classes são relevantes?

Vale a pena ver de que entidades fala o problema descrito:

“O laboratório **X** gere projectos de cientistas; cada projecto necessita de vários investigadores.”

Laboratório de Investigação

Que classes são relevantes?

Vale a pena ver de que entidades fala o problema descrito:

“O **laboratório** **X** gere **projectos** de **cientistas**; cada **projecto** necessita de vários **investigadores**”

Claramente, são entidades:

- o laboratório
- os projectos
- os investigadores/cientistas

O que deve guardar cada classe?

Que funcionalidades deve oferecer cada classe?

Laboratório de Investigação

Pretende-se implementar a classe que representa projectos (classe `Project`) e uma classe para iterar sobre colecções de projectos (classe `IteratorProjs`), considerando que um investigador é representado apenas pelo seu *email*.

A classe `Project` guarda a informação relevante dum projecto (número único, descrição, *email* do proponente, quantidade de investigadores colaboradores necessários e *emails* dos colaboradores efectivos). Pode-se saber o número do projecto, a sua descrição, quem foi o proponente e associar ou retirar colaboradores. Além disso, pode-se ordenar a lista dos colaboradores do projecto (lexicograficamente pelos *emails*).

Os objectos da classe `IteratorProjs` iteram sobre colecções de projectos, recebendo um vector de projectos e o número de projectos no vector. É possível saber se ainda há projectos para visitar e, caso haja, obter um projecto ainda não visitado. Os projectos devem ser obtidos (ou iterados) por ordem crescente do número do projecto.