

# DA1MCTA018-13SA - Programação Orientada a Objetos - Paulo Henrique Pisani - 2021.2

[Painel](#) / [Meus cursos](#) / [POO - DA1MCTA018-13SA - 2021.2](#) / [Classes, objetos, encapsulamento, UML](#) / [\[EP\] Lâmpada Inteligente](#)

Descrição

[Visualizar envios](#)

## [EP] Lâmpada Inteligente

**Data de entrega:** segunda, 14 Jun 2021, 23:59

**Arquivos requeridos:** GerenciadorLampadas.java ([Baixar](#))

**Tipo de trabalho:** Trabalho individual

Uma empresa está desenvolvendo um aplicativo para gerenciar lâmpadas inteligentes. Cada lâmpada inteligente é representada por uma instância da classe **LampadaInteligente** (essa classe já está implementada no sistema de correção automática e encontra-se no **pacote dispositivos**). Para este aplicativo, é necessário implementar a classe **GerenciadorLampadas**, que gerencia instâncias de lâmpadas inteligentes (ou seja, instâncias da classe LampadaInteligente).

A classe **LampadaInteligente** (que está no **pacote dispositivos**) possui os seguintes métodos:

- public void ligar(): liga a lâmpada;
- public void desligar(): desliga a lâmpada;
- public boolean getEstado(): retorna o estado da lâmpada (true=ligada, false=desligada).

A seguir são apresentados os métodos necessários na classe **GerenciadorLampadas**. Esses métodos podem ser usados pelo sistema de correção e por isso devem seguir a especificação apresentada no enunciado. Você pode incluir atributos e métodos auxiliares adicionais que considerar necessários durante a implementação.

Classe **GerenciadorLampadas**:

- public void adicionarLampada(): instancia uma nova lâmpada inteligente e adiciona a um vetor de LampadaInteligente. Considere que o tamanho máximo desse vetor é 10.
- public void ligarLampada(int indiceLampada): liga a lâmpada instanciada no índice informado. Antes de ligar a lâmpada, o método deve verificar se a lâmpada já está ligada. Se já estiver ligada, a lâmpada não deve ser ligada novamente. É possível saber se a lâmpada está ligada por meio do método getEstado() presente na classe LampadaInteligente.
- public void desligarLampada(int indiceLampada): desliga a lâmpada instanciada no índice informado. Antes de desligar a lâmpada, o método deve verificar se a lâmpada já está desligada. Se já estiver desligada, a lâmpada não deve ser desligada novamente. É possível saber se a lâmpada está ligada por meio do método getEstado() presente na classe LampadaInteligente.
- public void ligarTodasLampadas(): liga todas as lampadas. Lâmpadas já ligadas não devem ser ligadas novamente.
- public void desligarTodasLampadas(): desliga todas as lampadas. Lâmpadas já desligadas não devem ser desligadas novamente.

**Importante:** Submeta apenas a classe pública **GerenciadorLampadas** descrita no enunciado (a classe deve estar no pacote "sem nome", ou seja, não inclua a linha package no código). A classe submetida não pode realizar impressão de dados, utilizar java.util e System.out.

## Casos de teste

Formato dos casos de teste (que aparecem ao avaliar as classes no sistema de correção automática):

Entrada:

- operações no formato: [código operação] (índice lâmpada)
  - operação 1: adicionar lâmpada
  - operação 2: ligar lâmpada
  - operação 3: desligar lâmpada

- operação 4: ligar todas as lâmpadas
- operação 5: desligar todas as lâmpadas

Saída:

- métodos executados e saídas obtidas

◀ [EP] Porta

Seguir para...

[EP] Pedidos de produtos ▶

[VPL](#)



Este é o Ambiente Virtual de Aprendizagem da UFABC para apoio ao ensino presencial e semipresencial. Esta plataforma permite que os usuários (educadores/alunos) possam criar cursos, gerenciá-los e participar de maneira colaborativa.

## Informação

Conheça a UFABC

Conheça o NTI

Conheça o Netel

## Contato

Av. dos Estados, 5001. Bairro Bangu - Santo André /SP – Brasil. CEP 09210-580.

Siga-nos



[Universidade Federal do ABC](#) - [Moodle](#) (2020)

[Obter o aplicativo para dispositivos móveis](#)