

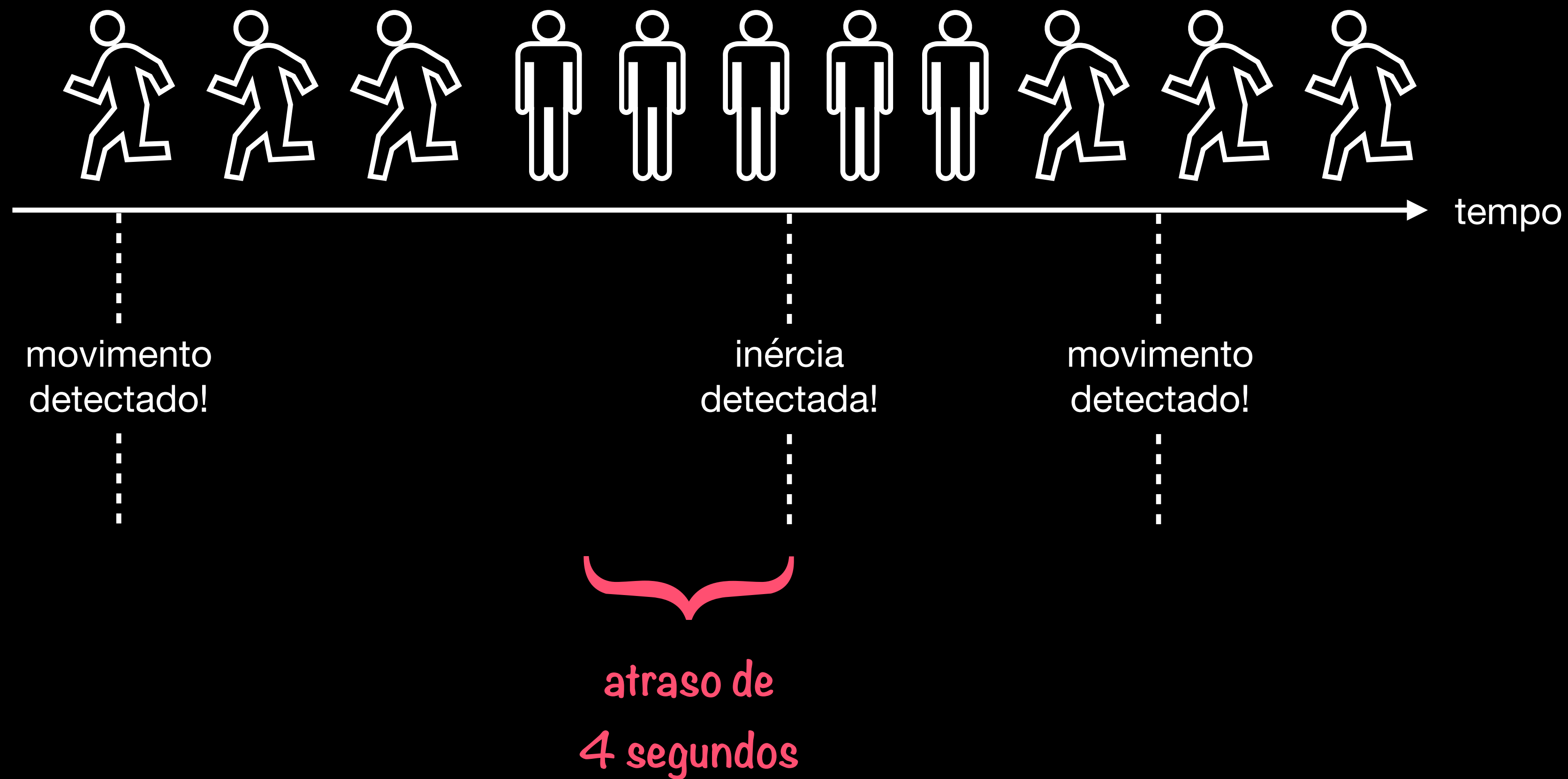
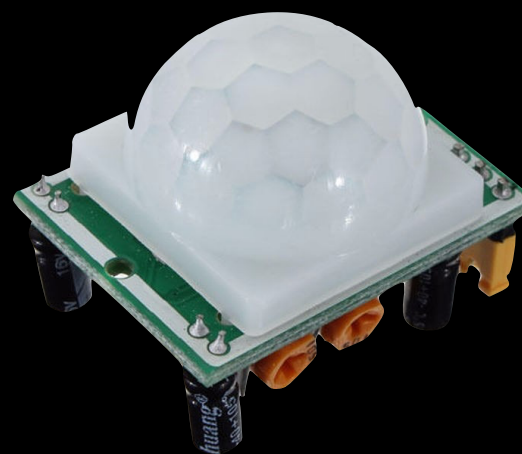
Projeto 05

Controle Automático – Prática

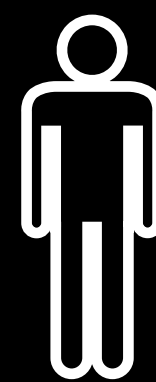
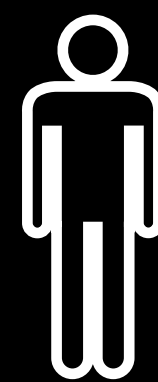
Jan K. S. – janks@puc-rio.br

ENG4033 – Projeto de Microcontroladores

Testes Iniciais



Atraso de Detecção nos Sensores de Movimento



:
inércia
detectada!

Como faz para
esperar mais tempo?



atraso de
4 segundos



atraso de mais alguns segundos

ao detectar **inércia**



inicie um Timer (global)
para esperar mais tempo

ao detectar **movimento**



cancele o Timer

Acréscimo no Tempo de Espera da Inércia

Crie um timer recorrente que imprima "olá" a cada 2 segundos.

Acenda o LED 1 ao detectar um movimento, e apague-o ao detectar a inércia.

↪ DICA: use `when_motion` e `when_no_motion`.



Testes Iniciais

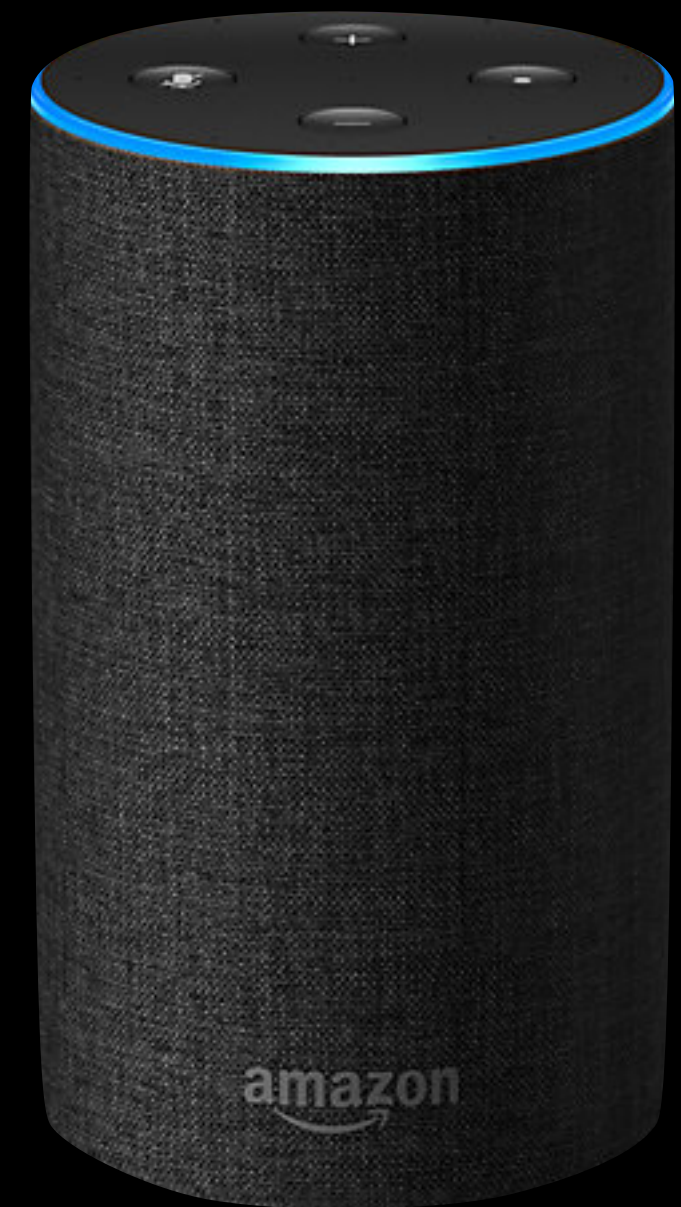
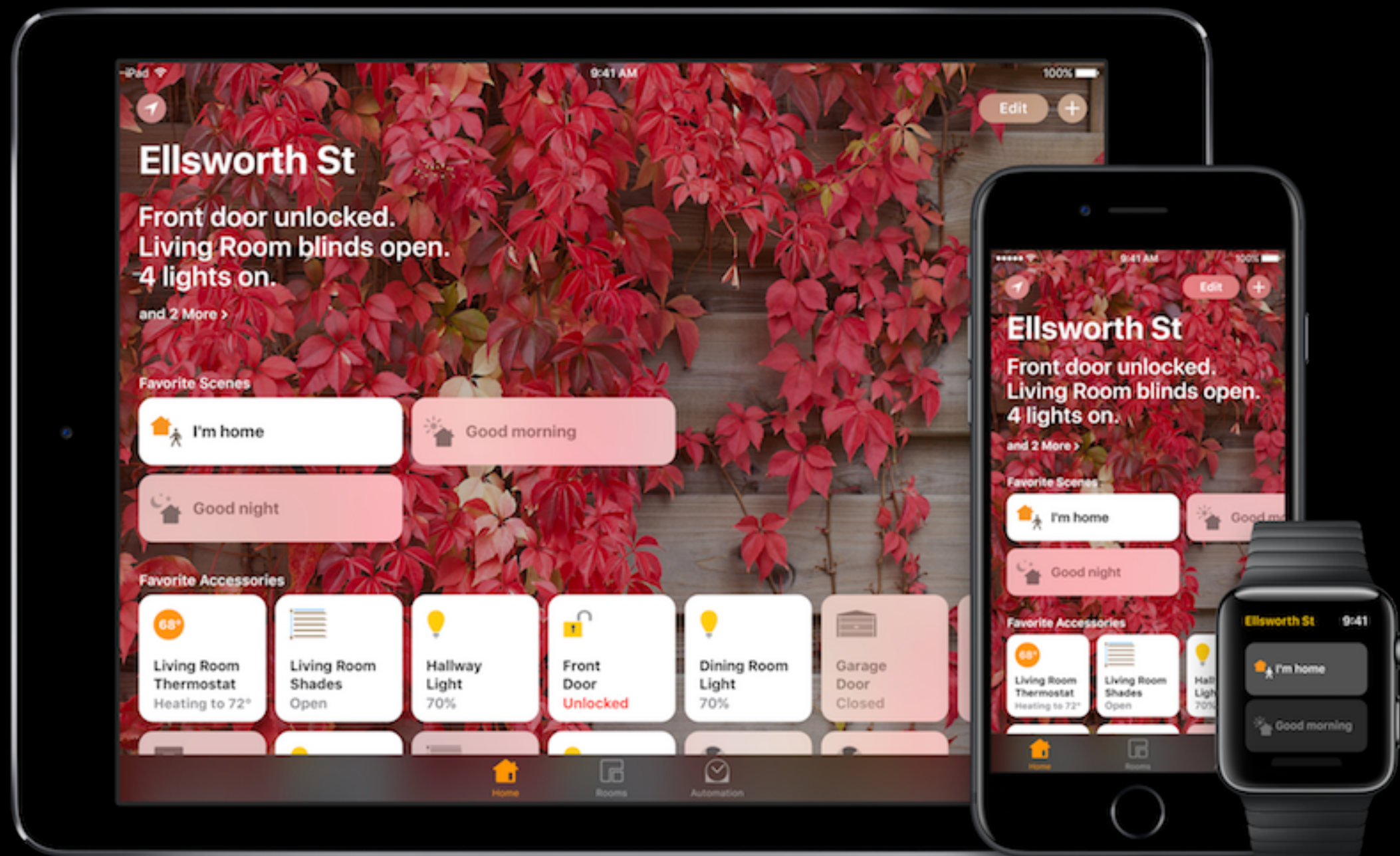
Acenda o LED 2 ao detectar um movimento e apague-o somente se não houver movimento por 8 segundos.

↪ DICA: veja o algoritmo do Timer no slide anterior. O LED 2 nunca pode estar apagado se o LED 1 estiver aceso.

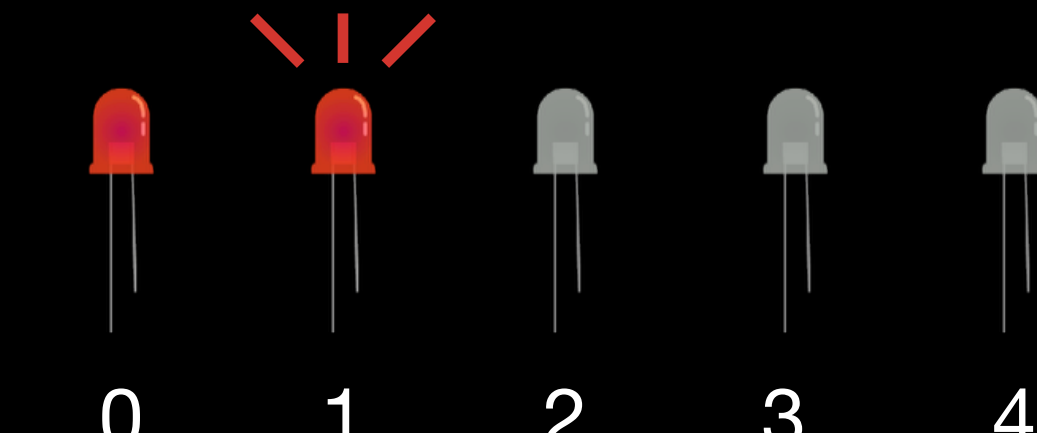
Crie um flow no Active Pieces com um trigger tipo **Webhook**. Copie o link do Webhook para o seu código. Ao apertar o Botão 1, faça um post para o link, enviando o valor do sensor de luz (em porcentagem) e o do sensor de distância (em cm). Teste o trigger.

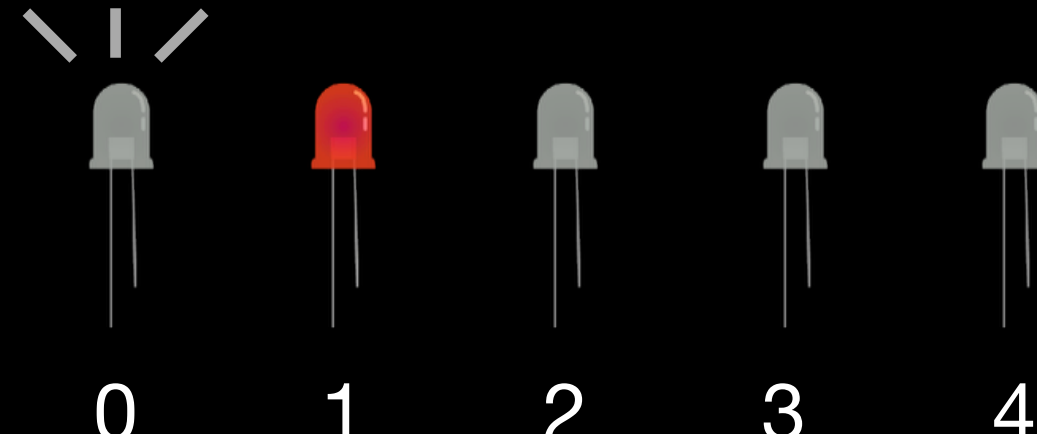
Adicione uma ação para escrever uma linha (append) num documento "Sensores" no Google Docs com os valores dos dois sensores, no formato "52% de luz / 13 cm". Publique o flow e teste.

Implementação

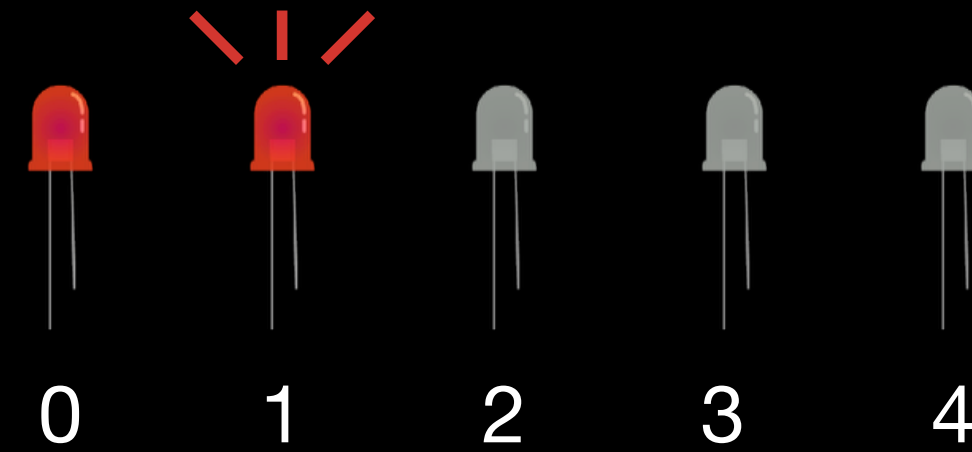


Smart Home

`atualiza_led(1, True)` → 

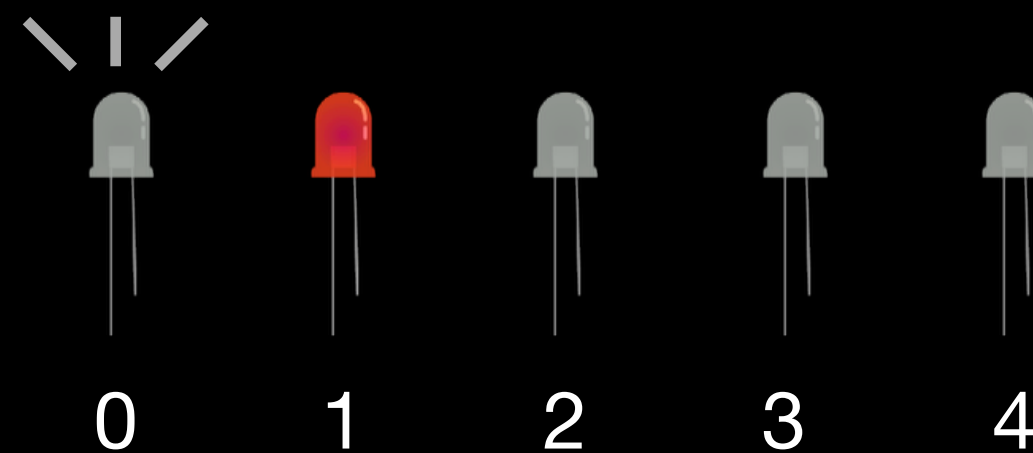
`atualiza_led(0, False)` → 

atualiza_led(1, True) →

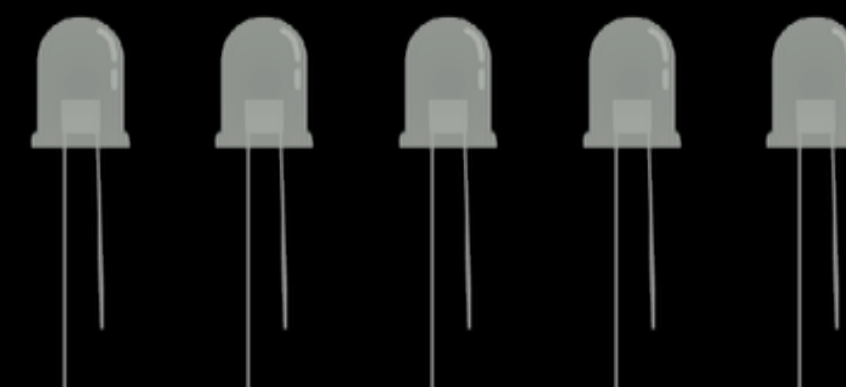
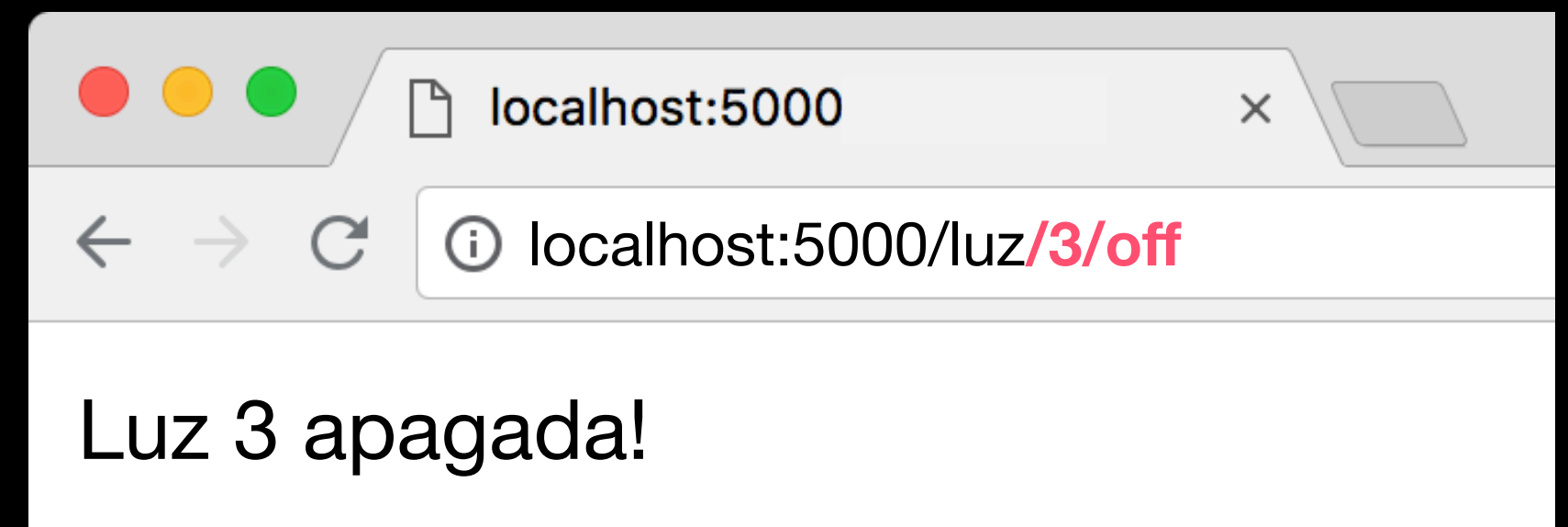
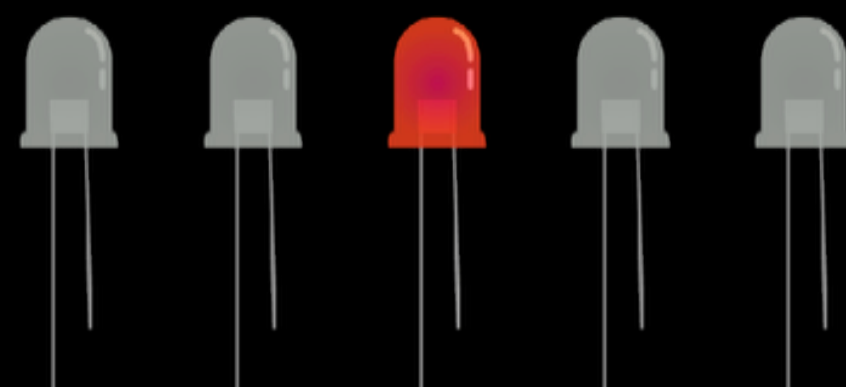
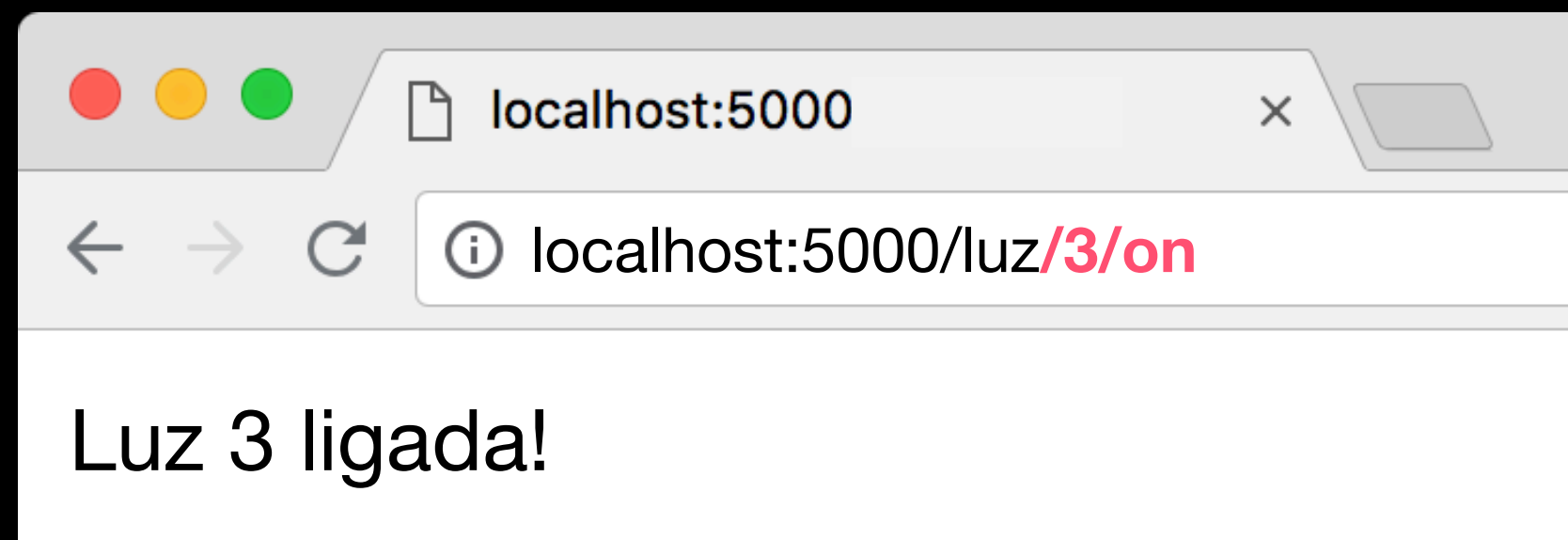


```
{  
  "data": datetime(2020, 8, 20, 15, 30, 45),  
  "estado_dos_leds": [True, True, False, False, False]  
}
```

atualiza_led(0, False) →



```
{  
  "data": datetime(2020, 8, 20, 15, 52, 27),  
  "estado_dos_leds": [False, True, False, False, False]  
}
```

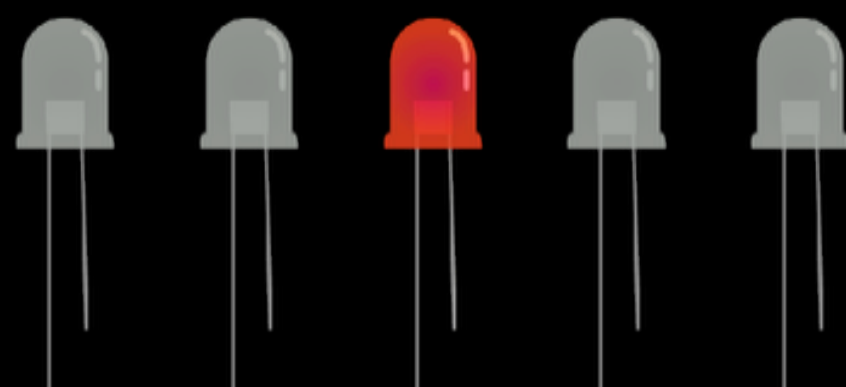


Parte 3: Controle de Luzes pelo Servidor

ativar luz



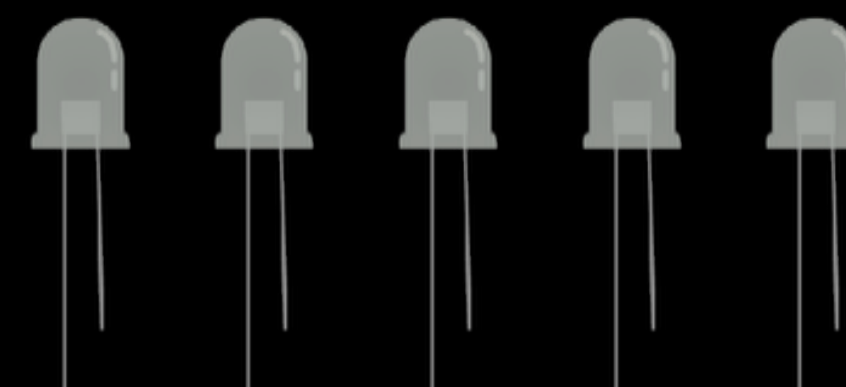
`http:// ... ngrok.io/luz/3/on`



desativar luz



`http:// ... ngrok.io/luz/3/off`





Atalhos (iOS)

ativar LED

desativar LED

Ativar LED

OK

Obter conteúdo de <https://757c-2804-7f1-e2c0-8e22-5981-f926-62fb-498f.ngrok-free.app/luz/3/>

on

Método

Cabeçalhos

Adicionar novo cabeçalho

Mostrar alerta Luz acesa

Título opcional

Mostrar Botão Cancelar

Desativar LED

OK

Obter conteúdo de <https://757c-2804-7f1-e2c0-8e22-5981-f926-62fb-498f.ngrok-free.app/luz/3/>

off

Método

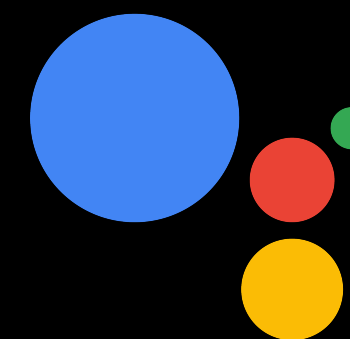
Cabeçalhos

Adicionar novo cabeçalho

Mostrar alerta Luz apagada

Título opcional

Mostrar Botão Cancelar



Google Assistente

Have a great afternoon

Get a Snapshot what's ahead for you

Get it

Search settings

POPULAR SETTINGS

Voice Match

Routines Multiple actions with one command

Personalization

Routines

New Routine to try

Workday Not enabled

Your routines

+ Add starter

Voice command

One non-voice starter available per Routine

Voice command

Time

This Routine will

Let's Get Ready now

Choose a phrase, like "Let's get ready"

When I say, "Hey Google..."

Let's Get Ready now

+ Add another

Done

When I say to my Assistant "Let's get ready"

+ Add starter

+ Add action

This Routine will

Add action

ADD

Enter a Google Assistant command

Abrir site <https://757c-2804-7f1-e2c0-8e22-5981-f926-62fb-498f.ngrok-free.app/luz/3/>

e.g. "set the volume to 50%" or "what's the weather"

Por algum motivo não funciona...



Comando de Voz no Celular



Implementação

Crie a função `atualiza_led` que recebe o índice do LED + o novo estado (True = aceso, False = apagado) e altere o estado desse LED. Chame a função para testar.

Dentro da função anterior, crie e insira um documento com a data/horário atual + lista com o estado (True/False) de cada LED em uma coleção do banco de dados. Teste a função novamente e verifique os dados em localhost:1234.

↳ DICA: preencha a lista usando `is_lit` dentro de um `for`.

Inicie um servidor Flask com uma página que receba o número do LED + o estado desejado (ex: `/led/1/on`) e atualize o LED solicitado. Teste a página em localhost:5000 e depois com o ngrok.

↳ DICA: use os parâmetros `int` e `string` na rota.

Crie dois atalhos / rotinas para chamar o seu servidor: um que acenda a luz 3 e outra que apague a luz 3. Chame-os por comando de voz.

↳ DICA: use o link do ngrok + o app Atalhos ou o Google Assistente.

Aperfeiçoamento



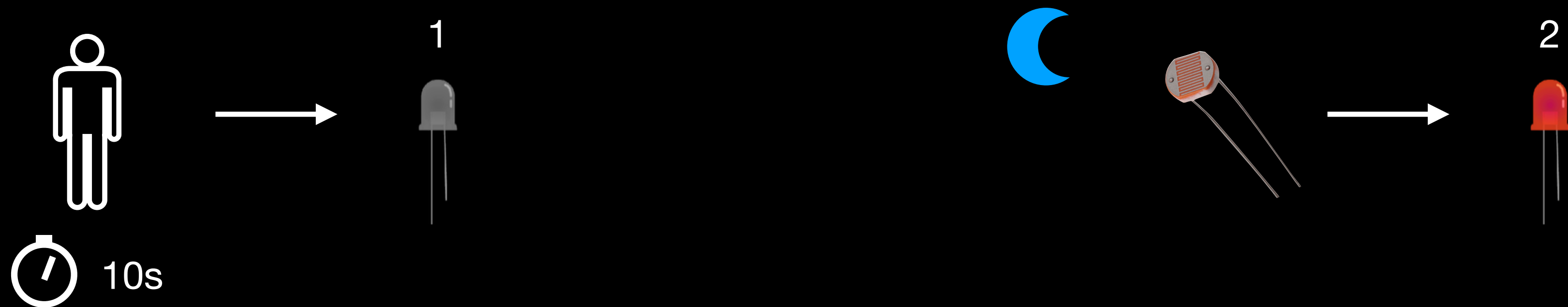
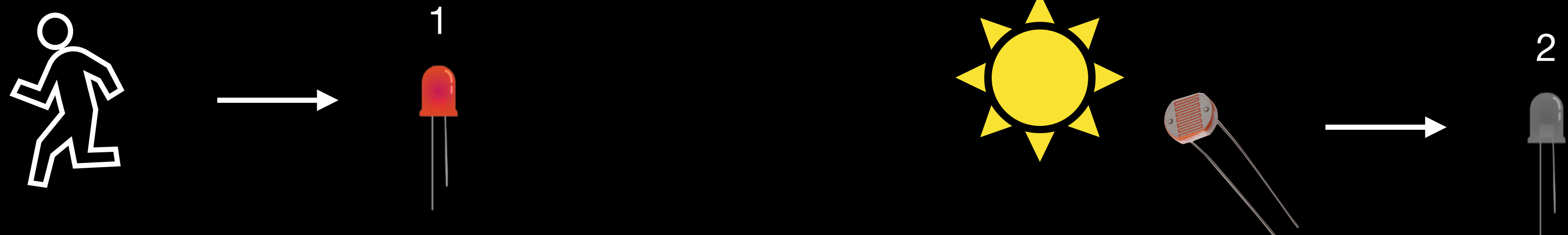
05b_implementacao.py

cópia
-----▶



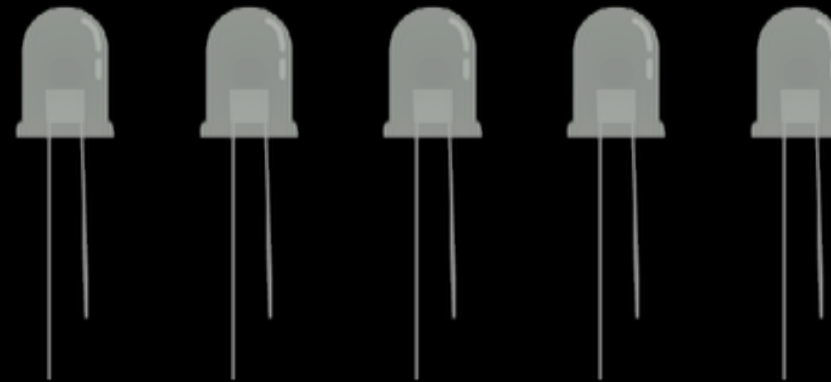
05c_aperfeicoamento.py

Cópia do Código da Implementação para o Aperfeiçoamento



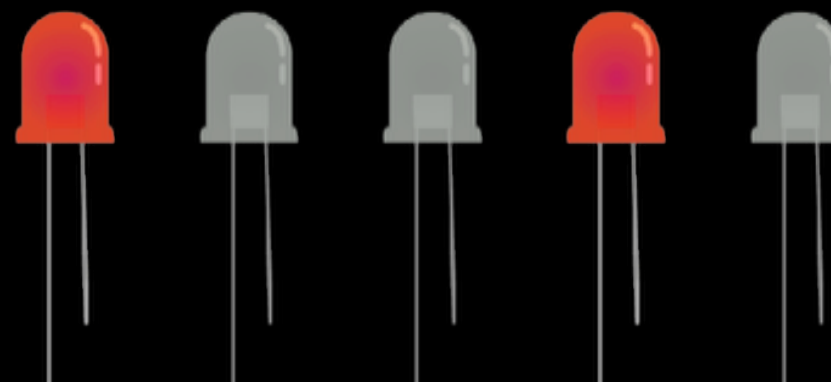
Controle dos LEDs 1 e 2 de Acordo com os Sensores

Início do Programa

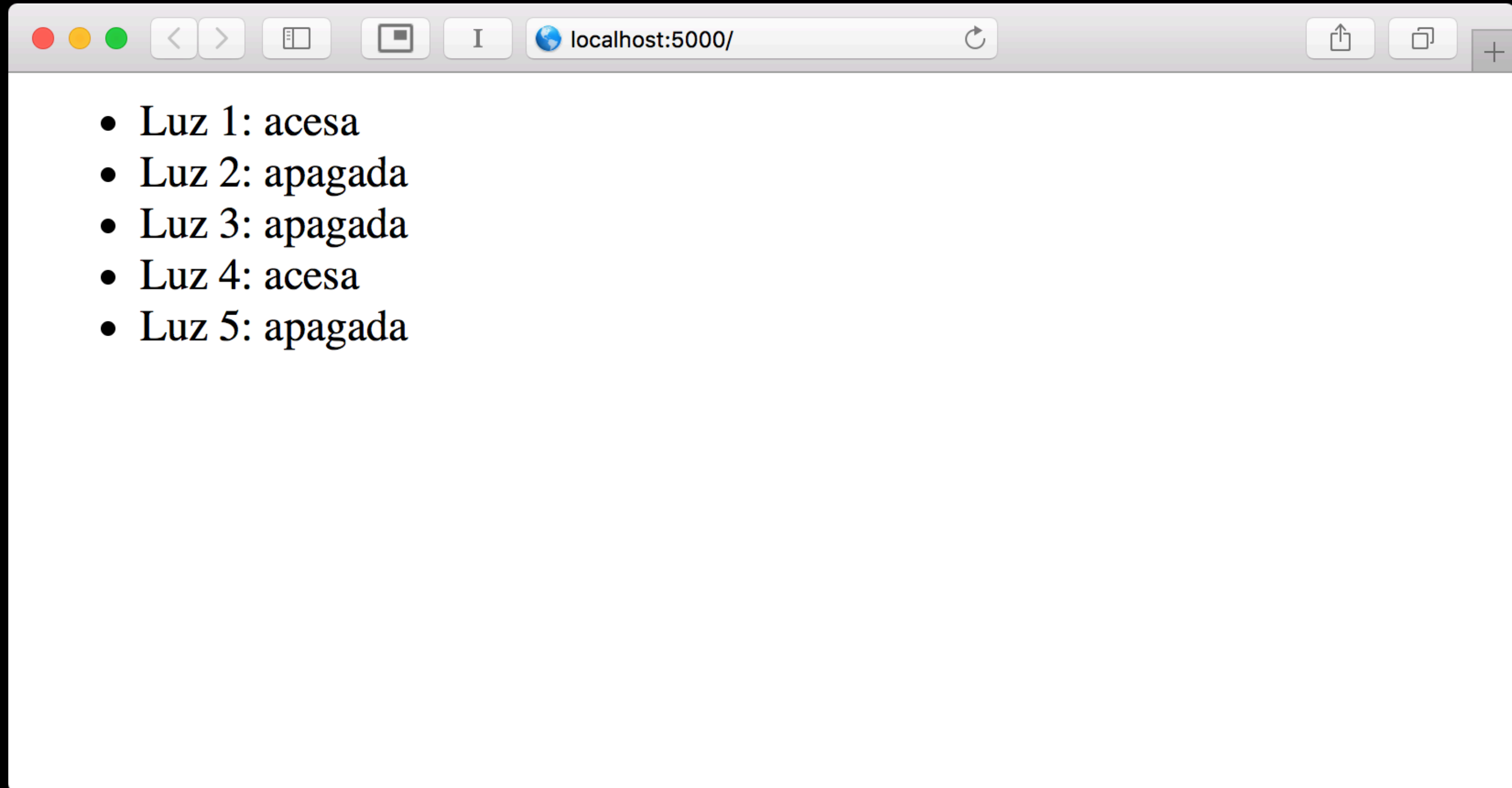


Último Dado:

```
{  
  "data": datetime(2020, 8, 20, 15, 30, 45),  
  "estado_dos_leds": [True, False, False, True, False]  
}
```



Restauração do Estado Anterior dos LEDs



Página com o Estado de Todos os LEDs



Aperfeiçoamento

Controle o LED 1 também de acordo com sensor de movimento: acenda-o quando houver movimento, e apague depois de 10 segundos sem movimento, chamando a função `atualiza_led` da Implementação.
↳ DICA: copie e adapte o código dos Testes Iniciais.

Controle o LED 2 de acordo com a luz captada: acenda-o caso esteja escuro e apague-o caso esteja claro, chamando a função `atualiza_led` da Implementação.
↳ DICA: use `when_dark` / `when_light`.

Quando o programa iniciar, busque o dado mais recente no banco e restaure o estado anterior dos LEDs.
↳ DICA: você pode passar um critério de busca vazio + ordenação para a `find_one`.

Crie uma página que verifique e exiba o estado de todos os LEDs, seguindo o formato HTML dos slides anteriores.

↳ DICA: gere o HTML dentro da função, concatenando strings dentro de um `for`.

Desafio Extra



05c_aperfeicoamento.py

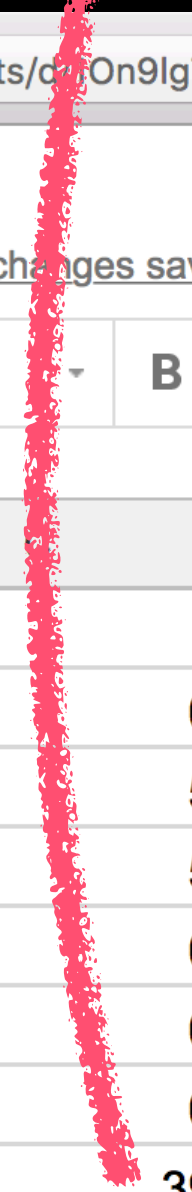
cópia
-----▶



05d_desafio.py

Cópia do Código do Aperfeiçoamento para o Desafio

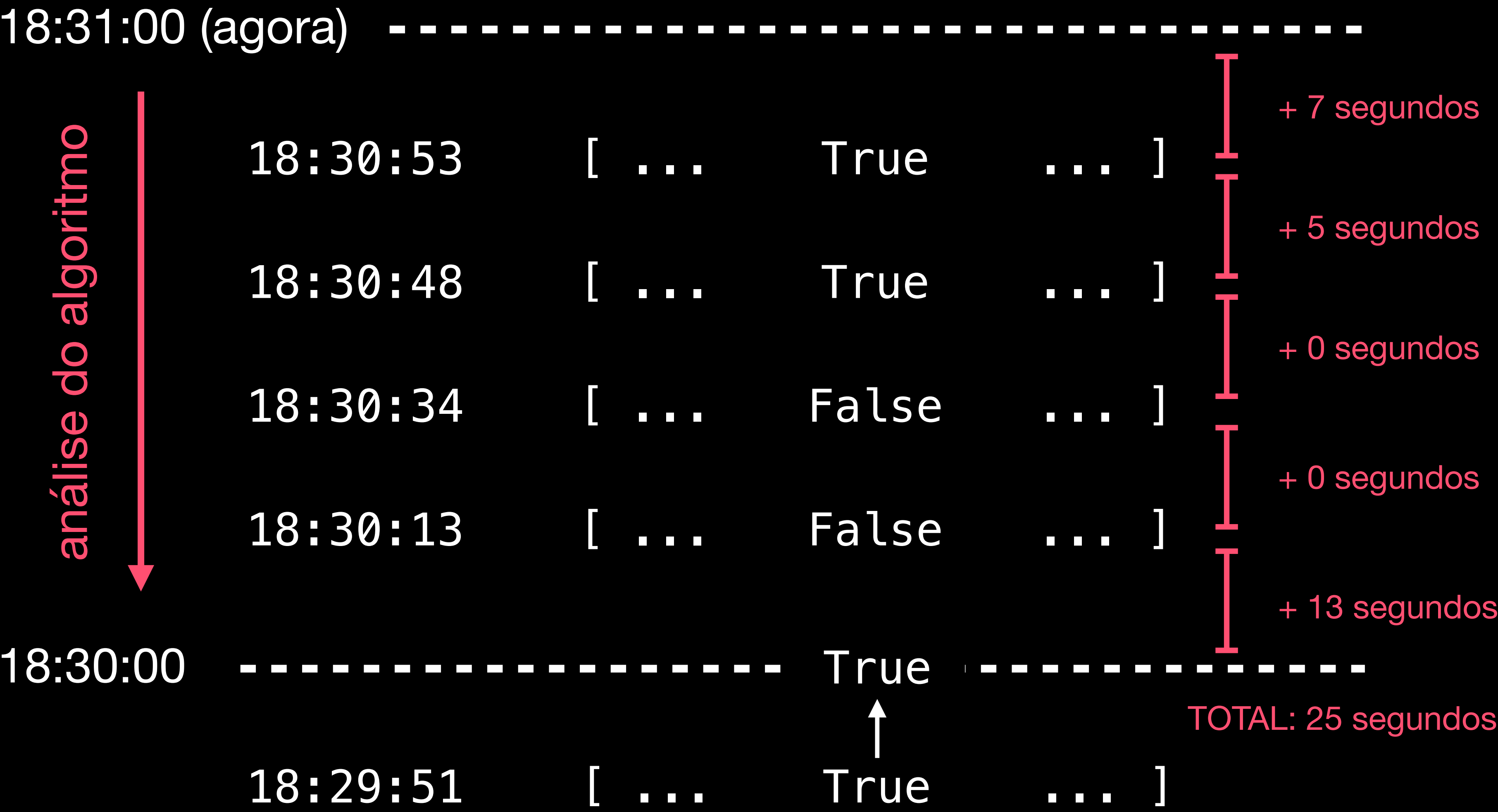
entre 12:18:00 até 12:18:59,
Luz 3 ficou acesa durante 39 segundos



	A	B		D	E	F
5	Data	Luz 1	Luz 2	Luz 3	Luz 4	Luz 5
6	September 17, 2018 at 12:12PM	0	0	0	0	0
7	September 17, 2018 at 12:13PM	8	5	4	4	0
8	September 17, 2018 at 12:14PM	8	5	4	8	3
9	September 17, 2018 at 12:15PM	0	0	0	0	0
10	September 17, 2018 at 12:16PM	0	0	0	0	0
11	September 17, 2018 at 12:17PM	0	0	10	13	0
12	September 17, 2018 at 12:18PM	51	39	18	0	6
13	September 17, 2018 at 12:19PM	1	1	0	0	3
14	September 17, 2018 at 12:21PM	0	0	0	0	0
15	September 17, 2018 at 12:21PM	29	0	0	0	0
16	September 17, 2018 at 12:22PM	29	0	0	3	2
17	September 17, 2018 at 12:23PM	0	0	0	3	2
18	September 17, 2018 at 12:24PM	0	0	0	0	0

Planilha com Resumo do Tempo Total de Iluminação de Cada Luz

LED N



Exemplo de Contagem do Tempo a Partir de Vários Dados



Desafio Extra

Crie uma função que receba o número de um LED e uma data/horário, **retornando total de segundos em que o LED ficou aceso após aquela data**. Teste a função no shell do Thonny.

↳ DICA: use as funções `find` / `find_one`, os operadores `$lt` / `$gt` e a ordenação decrescente.

Crie uma função que obtenha o tempo aceso dos 5 LEDs desde 1 minuto atrás e **escreva esses valores + a data em uma nova linha na planilha do Google Sheets**. Chame a função para testar uma vez.

Crie um Timer recorrente que chame a função acima a cada 30 segundos. Teste o resultado acendendo e apagando alguns LEDs por um tempo.

Possíveis Melhorias



Possíveis Melhorias

Como melhorar a detecção de luz de modo que os eventos de claridade e escuridão não sejam chamados erraticamente quando a luz está muito próxima do limiar?



janks.link/micro/projeto05.zip

Material do Projeto 05