

## Avaliação

**Olá! Obrigado por nos mandar seu currículo!!**

Na Data Machina você enfrentará diferentes desafios, que testam desde a sua organização até problemas de programação. Para isso, preparamos um desafio para avaliarmos suas habilidades e testar sua capacidade em resolver problemas, que muitas vezes podem ser inéditos, portanto “*Google it*”.

Para podermos te avaliar, você passará por um desafio. O desafio consiste em desenvolver algumas linhas de **código em Python** para resolver dois problemas independentes. A execução do problema deverá ser feita via uma API local também desenvolvida por você.

### Problema 1.

Você deverá criar um código que calcule o n-ésimo termo da sequência de Fibonacci. Para calcular o n-ésimo termo você **não pode incluir** bibliotecas externas ou até mesmo funções built-in de Python.

Avaliaremos a resposta da função de Fibonacci através de uma chamada de API que deverá receber apenas um único parâmetro “n”, e deve retornar o valor calculado de Fibonacci(n) na resposta da API.

### Problema 2.

Agora que você já é capaz de criar uma API que recebe parâmetros e retorna algum resultado de algum processo interno, dificultaremos um pouco o problema.

Nesse caso, você tem o seguinte desafio, as plataformas Lala e Ogi oferecem serviço de entrega, cada uma delas possui alguns veículos e suas restrições de tamanho e peso como mostra a tabela abaixo:

Plataforma	Veículo	Largura máx.	Altura máx.	Espessura máx.	Peso máx.
Lala	Moto	35 cm	40 cm	30 cm	20 kg
	Fiorino	188 cm	133 cm	108 cm	500 kg
	Carreto	300 cm	180 cm	200 cm	1500 kg
Ogi	Moto	52 cm	36 cm	52 cm	20 kg
	SUV	125 cm	80 cm	60 cm	200 kg

O seu desafio é fazer uma função que recebe uma lista de itens que contenha nas características de cada item seu peso, largura, espessura e a altura. E deve determinar **para cada plataforma** qual deve ser o veículo ideal para transportar esses itens (se os

itens couberem em pelo menos um veículo). Vale ressaltar que o preço da entrega aumenta conforme aumenta a capacidade do veículo (mesmo não estando totalmente preenchido).

Para resolver esse problema você **pode incluir** bibliotecas prontas e toda a sua criatividade. E assim como dito no problema 1, para testarmos o problema faremos uma chamada na API criada, passando como parâmetros uma lista de itens com as características necessárias para resolver o problema.

### **Extra**

Agora que você sabe criar uma API e resolver um problema razoavelmente complexo com ela, você tem em suas mãos algo bastante valioso que poderia ser acessado a partir de qualquer máquina. Mas você não quer que qualquer um acesse, será que você seria capaz de **tornar sua API mais segura**?

### **Entrega / Avaliação**

Você terá **aproximadamente uma semana para enviar** os seus resultados, portanto faça com calma. Todo o código desenvolvido no desafio deve ser adicionado em um repositório em seu GitHub, e assim que finalizá-lo, nos enviar o link do repositório nos seguintes emails: [mcabral@datamachina.com.br](mailto:mcabral@datamachina.com.br) e [lucas@datamachina.com.br](mailto:lucas@datamachina.com.br). O repositório deve conter um README explicando como utilizar a API criada para resolver os problemas.

Nesse desafio avaliaremos a qualidade de organização/documentação do repositório e do seu código, como também o raciocínio utilizado para resolver os problemas. Lembramos que o código deve ser escrito em **linguagem Python** tanto para as funções que envolvem a solução dos desafios como também para a API que executa as funções.

Após a entrega, marcaremos uma data para que você faça uma breve apresentação sobre o que foi o código desenvolvido, dificuldades encontradas e escolhas feitas.