

# Computação 1, 2021.1

## Lista 2

Data limite para entrega: 03/08 às 23:59

Submeta suas soluções colocando os arquivos correspondentes na sua pasta do Google Drive\*

### Parte 1 – Obrigatória

**Questão 1.** Usando o módulo `math` e qualquer outro que julgar apropriado, faça programas de `python` que calculem cada uma das funções matemáticas abaixo:

a.  $f(x) = \sqrt{x + x^2}$ .

b.  $f(x, y) = \begin{cases} \log_2(x), & \text{se } x \text{ for positivo e ímpar} \\ \text{magia}(y), & \text{caso contrário,} \end{cases}$

sendo `magia` uma função (matemática) calculada por uma função de `python` de mesmo nome, disponível em um *módulo* (fictício) chamado `funcoes`.

**Questão 2.** Para cada programa de `python` abaixo, desenhe a árvore dos quadros criados ao longo da execução do programa todo:

a. 

```
def f():  
    g()
```

```
def g():  
    return 1
```

```
f()
```

b. 

```
def f():  
    return 3
```

```
def g():  
    return 4
```

```
f()
```

---

\*Link recebido por email em 19/7/2021 — o nome é parecido com <seu nome> - Computação 1 - Submissões e Feedback.

```

c. def f():
    x = 1
    def g(n):
        print(x+1)
        return h(n)
    return g(3*x)

def h(n):
    return x+n

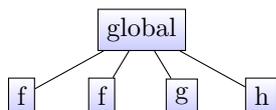
x = 3
f()

```

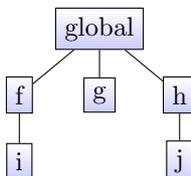
**Questão 3.** Agora, para cada árvore de quadros abaixo, escreva um programa de `python` que gere em seu ambiente exatamente a estrutura de quadros descrita ao longo de sua execução. Atenção! Você pode fazer no máximo **uma** chamada de função no quadro `global`, e os únicos comandos no seu programa devem ser definições de funções.



a.



b. e



c.

**Questão 4.** Faça uma função de `python` que receba como entrada três números  $a$ ,  $b$  e  $c$ , e retorne quantas soluções reais possui a equação  $ax^2 + bx + c = 0$ .

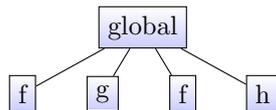
**Questão 5. Caçadores de asteróides** Um asteróide está passando pela Terra à velocidade  $V_a$  (relativa à Terra). Um foguete será lançado ao seu encalço, saindo da Terra com velocidade  $V_f$ . O asteróide está acelerando com aceleração  $A_a$ , e o foguete tem motores que dão uma aceleração máxima igual a  $A_f$ .

Escreva uma função que recebe estes quatro argumentos (todos eles `float`), e retorna

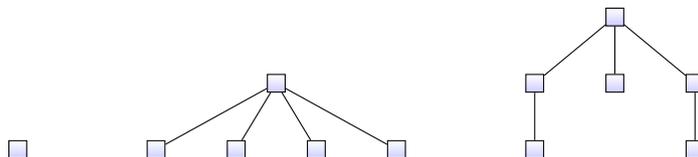
- `True` se o foguete pode ficar mais rápido do que o asteróide; e
- `False` em caso contrário.

## Parte 2 — Desafio opcional

**Questão 6.** Escreva um programa de `python` que gere em seu ambiente exatamente a estrutura de quadros descrita. Atenção! Você pode fazer no máximo **uma** chamada de função no quadro `global`, e os únicos comandos no seu programa devem ser definições de funções.



**Questão 7.** Vamos chamar de *formato* de uma árvore de quadros a árvore resultante de se remover os rótulos de cada nó. Por exemplo, os formatos das árvores de quadros da questão 4 são, respectivamente:



É verdade que para *qualquer* árvore sem rótulos e não-vazia  $T$  existe um programa de `python` cujo ambiente tem uma árvore de quadros com formato exatamente igual a  $T$ ? Se sim, dê o melhor argumento que puder justificando sua resposta. Se não, dê um contraexemplo e um argumento justificando o fato dele ser um contraexemplo. (Aqui, uma *árvore sem rótulos* é simplesmente uma coleção qualquer de caixas ligadas por traços, como nas figuras acima, de forma que não haja nenhum ciclo fechado na figura mas que entre qualquer par de caixas sempre exista um caminho de uma à outra passando pelos traços.)