

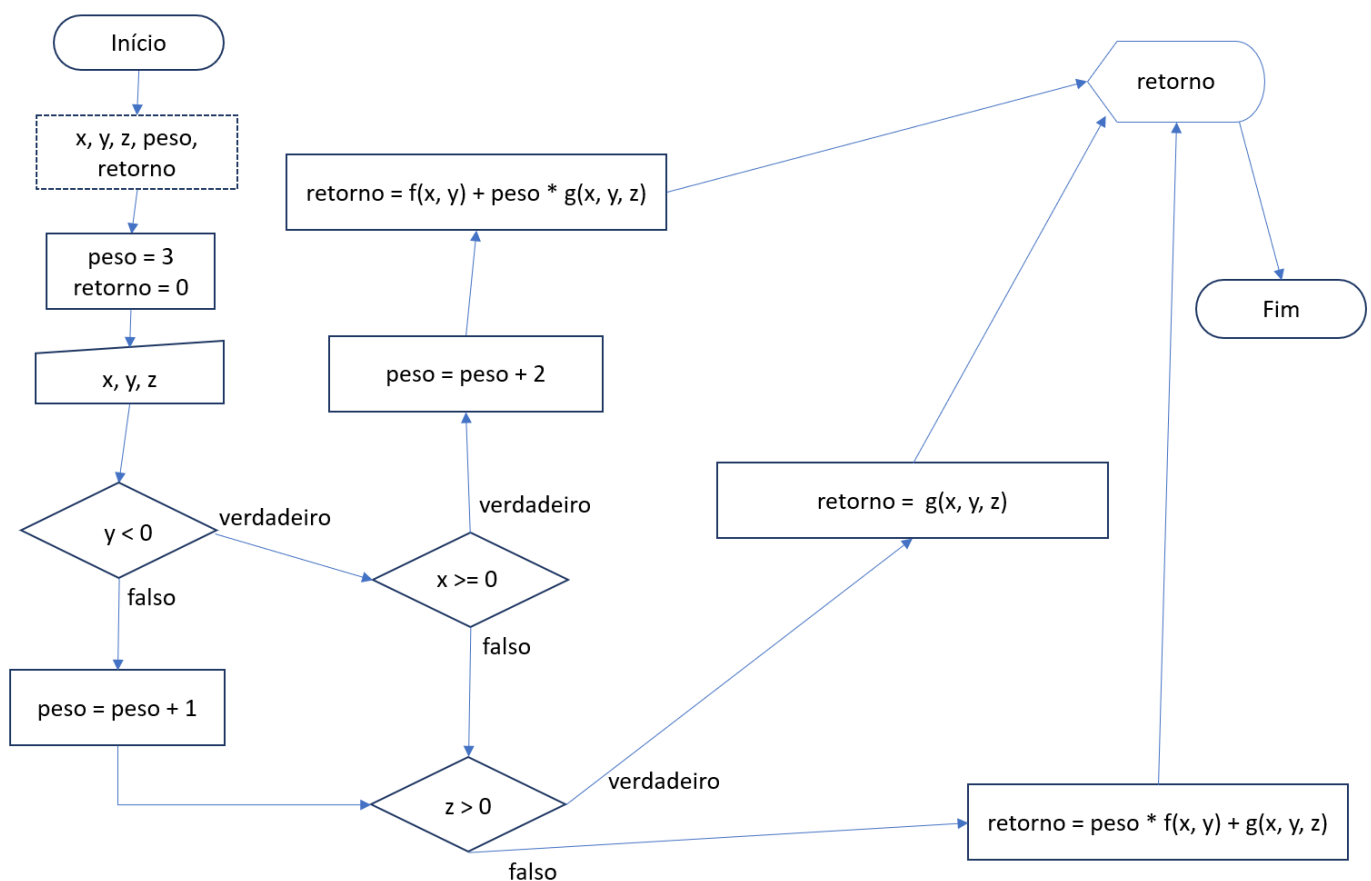
Nome: _____ Matrícula: _____ IP: _____

Você recebeu o enunciado do Teste 2 da G1 que deve ser **resolvido em sala de aula e enviado pela plataforma EAD**.

O teste contém três questões. Ele é individual, tem a duração de **50 minutos** e todas as atividades relacionadas à solução do trabalho proposto devem ser realizadas, respeitando-se o código de ética do CTC disponível na plataforma EAD.

Para cada questão do teste deverá ser criado um script correspondente que deverá ser salvo com o nome “INF1514_TEAG1_QX_MATRICULA.R”, substituindo o texto “MATRICULA” pelo número da sua matrícula e X pelo número da questão. Cada script deverá conter todas as implementações realizadas para a correspondente questão, incluindo os testes, sendo que a criação e organização dos scripts faz parte da resolução do trabalho.

Considere o seguinte fluxograma criado por um analista para processar dados coletados de um experimento.



Sabendo que $f(x, y) = x^3y - 3x^2y^2 - 4.6x$ e $g(x, y, z) = x^3y^2 - 5x^2z^2 + 3yz$ são funções que a partir dos parâmetros de entrada retornam um valor calculado segundo as respectivas fórmulas, faça:

1) Em um script R implemente uma função chamada **calculaF**, que retorna o valor da $f(x, y)$ e uma função chamada **calculaG** que retorna o valor da $g(x, y, z)$. A seguir, no mesmo script, implemente um algoritmo que apresenta na tela os valores retornados pelas funções **calculaF** e **calculaG** dados como parâmetros a função: **calculaF**, os valores **5** e **-2** para x e y , respectivamente; **calculaG** os valores **-3**, **-2** e **4** para x , y e z , respectivamente. (2,0 pontos)

2) Elabore um script em R que cria os seguintes vetores:

- vetorX, contendo **7** elementos iniciados em **-2.5**, com intervalo de **0.8**;
- vetorY, contendo os elementos **3, -2, 6, 8, 3, 2, -1**, nesta ordem;
- vetorZ, contendo uma sequência de valores entre **-3** e **3**, com intervalo de **1**.

A seguir, no mesmo script, faça uso de um dos comandos do R que permitem criar ciclos para chamar a função **calculaG** utilizando como parâmetros trincas de dados (**x, y, z**) formadas pelos elementos dos vetores vetorX, vetorY e vetorZ. O script deve apresentar na tela o menor valor e a média, nesta ordem, dos valores retornados pela função **calculaG** para cada uma das trincas de dados. Observe que a primeira trinca de dados (**x, y, z**) possui os valores **(-2.5, 3, -3)** e a segunda os valores **(-1.7, -2, -2)**. (4,0 pontos)

4) Com base no fluxograma, escreva uma função chamada **processaFluxograma** que recebe como parâmetros os valores que deveriam, segundo o fluxograma, ser fornecidos pelo usuário, e, em vez de apresentar na tela o resultado do processamento, este seja retornado pela função. Teste a função utilizando os valores **2, -3** e **-1** para **x, y** e **z**, respectivamente. (4,0 pontos)

Solução:

Questão 1

```
calculaF <- function(x, y)

{

  retorno <- (x^3) * y - 3 * (x^2) * (y^2) - 4.6 * x

  return(retorno)

}

calculaG <- function(x, y, z)

{

  retorno <- (x^3) * (y^2) - 5 * (x^2) * (z^2) + 3 * y * z

  return(retorno)

}

#variáveis

valorX <- 0.0

valorY <- 0.0

valorZ <- 0.0

valor <- 0.0

#inicio

valorX <- 5

valorY <- -2

valor <- calculaF(valorX, valorY)

print(valor)

valorX <- -3

valorY <- -2

valorZ <- 4
```

```
valor <- calculaG(valorX, valorY, valorZ)
```

```
print(valor)
```

```
#fim
```

Questão 2

```
calculaG <- function(x, y, z)
```

```
{
```

```
  retorno <- (x^3) * (y^2) - 5 * (x^2) * (z^2) + 3 * y * z
```

```
  return(retorno)
```

```
}
```

```
#variáveis
```

```
contador <- 0
```

```
vetorX <- NULL
```

```
vetorY <- NULL
```

```
vetorZ <- NULL
```

```
vetorValor <- NULL
```

```
#inicio
```

```
vetorX <- seq(-2.5, by = 0.8, length = 7)
```

```
vetorY <- c(3, -2, 6, 8, 3, 2, -1)
```

```
vetorZ <- -3:3
```

```
vetorValor <- rep(0, length(vetorX))
```

```
while (contador < length(vetorX))
```

```
{
```

```
  contador <- contador + 1
```

```
  vetorValor[contador] <- calculaG(vetorX[contador], vetorY[contador], vetorZ[contador])
```

```
}
```

```
print(min(vetorValor))
```

```
print(mean(vetorValor))
```

```
#fim
```

Questão 3

```
calculaF <- function(x, y)
```

```
{
```

```
  retorno <- (x^3) * y - 3 * (x^2) * (y^2) - 4.6 * x
```

```
  return(retorno)
```

```
}
```

```
calculaG <- function(x, y, z)
```

```
{
```

```
  retorno <- (x^3) * (y^2) - 5 * (x^2) * (z^2) + 3 * y * z
```

```
  return(retorno)
```

```
}
```

```
processaFluxograma <- function (x, y, z)
```

```
{
```

```
  retorno <- 0.0
```

```
  peso <- 3
```

```
  if (y < 0) {
```

```
    if (x >= 0) {
```

```
      peso = peso + 2
```

```
      retorno <- calculaF(x, y) + peso * calculaG(x, y, z)
```

```
} else {  
  
  if (z > 0) {  
  
    retorno <- calculaG(x, y, z)  
  
  } else {  
  
    retorno <- peso * calculaF(x, y) + calculaG(x, y, z)  
  
  }  
  
}  
  
} else {  
  
  peso = peso + 1  
  
  if (z > 0) {  
  
    retorno <- calculaG(x, y, z)  
  
  } else {  
  
    retorno <- peso * calculaF(x, y) + calculaG(x, y, z)  
  
  }  
  
}  
  
return(retorno)  
  
}  
  
#variáveis  
  
x <- 0.0  
  
y <- 0.0  
  
z <- 0.0  
  
valor <- 0.0  
  
#inicio  
  
x <- 2
```

```
y <- -3
```

```
z <- -1
```

```
valor <- processaFluxograma(x, y, z)
```

```
print(valor)
```

```
#fim
```