MAP 2112 – Introdução à Lógica de Programação e Modelagem Computacional

1º Semestre - 2020

Prof. Dr. Luis Carlos de Castro Santos

lsantos@ime.usp.br

NÃO ESQUEÇA DE INICIAR A GRAVAÇÃO

MAP 2112 – Introdução à Lógica de Programação e Modelagem Computacional

1º Semestre - 2020

Prof. Dr. Luis Carlos de Castro Santos

lsantos@ime.usp.br

ALGUNS PROBLEMAS DE PROVAS PRESENCIAIS ANTERIORES

Questão 1 (2 pts) - Escreva um script em R para que dados dois vetores x e y de mesmo comprimento produza um vetor z cujas componentes são a soma das componentes de x e y se o valor da componente de x é maior que a de y caso contrário a soma é zero.

Questão 1 (2 pts) - Escreva um script em R para que dados dois vetores x e y de mesmo comprimento produza um vetor z cujas componentes são a soma das componentes de x e y se o valor da componente de x é maior que a de y caso contrário a soma é zero.

Criar um índice que evolui por exemplo i de 1 a n

Evoluir esse índice – necessidade de laço: for

Comparar as componentes de x e y (no caso x[i] e y[i]) e calcular z[i] de acordo

```r
# exemplo
x <- c(1,2,3,4,5)
y <- c(3,3,3,3,3)

n <- length(x)
z <- rep(0,n)

for (i in 1:n) {
  if (x[i]>y[i])
    {z[i] <- x[i] + y[i]}
      else
    {z[i] <- 0}
}
print(z)
```

Questão 3 (3 pts). A Série de Taylor é uma aproximação polinomial de funções contínuas, por exemplo:

$$\mathrm{sen}(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \frac{x^{11}}{11!} + \cdots$$

Escreva um script em R para encontrar quantos termos são necessários para aproximar a função seno com precisão menor 10^-8.

Questão 3 (3 pts). A Série de Taylor é uma aproximação polinomial de funções contínuas, por exemplo:

$$\operatorname{sen}(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \frac{x^{11}}{11!} + \cdots$$

Escreva um script em R para encontrar quantos termos são necessários para aproximar a função seno com precisão menor 10^-8.

É necessária uma função que implemente o fatorial:

n ! = n*(n-1)*(n-2)...*2.1

A soma só computa as potências ímpares

Os sinais dos termos se invertem:

3 é negativo, 5 é positivo,7 é negativo, 9 é positivo e assim por diante

O maior valor que a função seno assume é 1, para x = pi/2

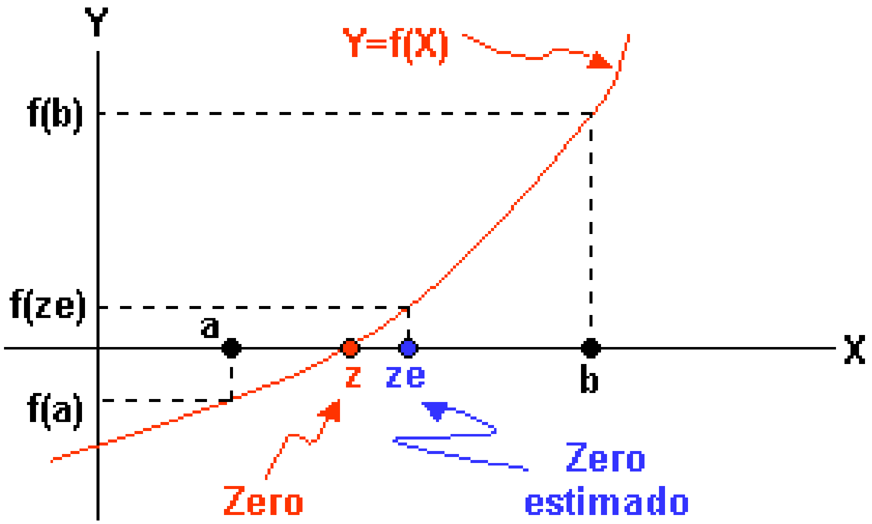
A soma deve terminar quando o termo a ser adicionado for menor que 10^(-8)

Deve ser necessário um laço que incrementa o termo até a tolerância ser atingida

```r
fatorial <- function(n){
     fat <- 1
     for (i in 1:n){fat <- fat*i}
     fat
}
i <- 1
tol <- 1.E-8
termo <- 1.
x <- pi/2.
senx <- x
while(abs(termo) > tol){
     n <- 2*i + 1
     termo <- ((-1)^(i))/fatorial(n)*(x^(n))
     senx <- senx + termo
     i <- i + 1
     print(i)
     print(termo)
     print(senx)
     print(sin(pi/2))
}
```

MÉTODO DA BISSEÇÃO

- A estimativa do zero da função **Y=f(X)** é feita a partir do ponto médio do intervalo analisado.
- Se o valor estimado não atender à tolerância estabelecida para o problema, ou seja, **|f(ze)|>tol**, redefine-se o intervalo de estudo, repetindo-se a estratégia até que a tolerância seja verificada.



Equação de recorrência:

$$ze = \frac{a + b}{2}$$

Questão 4 (3 pts). Escreva o script em R que implementa o algoritmo do método da bisseção descrito abaixo.

Algoritmo para o método da bissecção. (Ruggiero e Lopes, 1997.)

Seja f(x) contínua em [a, b] e tal que f(a)f(b) < 0.

1) Dados iniciais:
   a) intervalo inicial [a, b]
   b) precisão ε
2) Se $(b - a) < \varepsilon$, então escolha para $\bar{x}$ qualquer $x \in [a, b]$. FIM.
3) k = 1
4) M = f(a)
5) $x = \dfrac{a + b}{2}$
6) Se $Mf(x) > 0$, faça a = x. Vá para o passo 8.
7) b = x
8) Se $(b - a) < \varepsilon$, escolha para $\bar{x}$ qualquer $x \in [a, b]$. FIM.
9) k = k + 1. Volte para o passo.5.

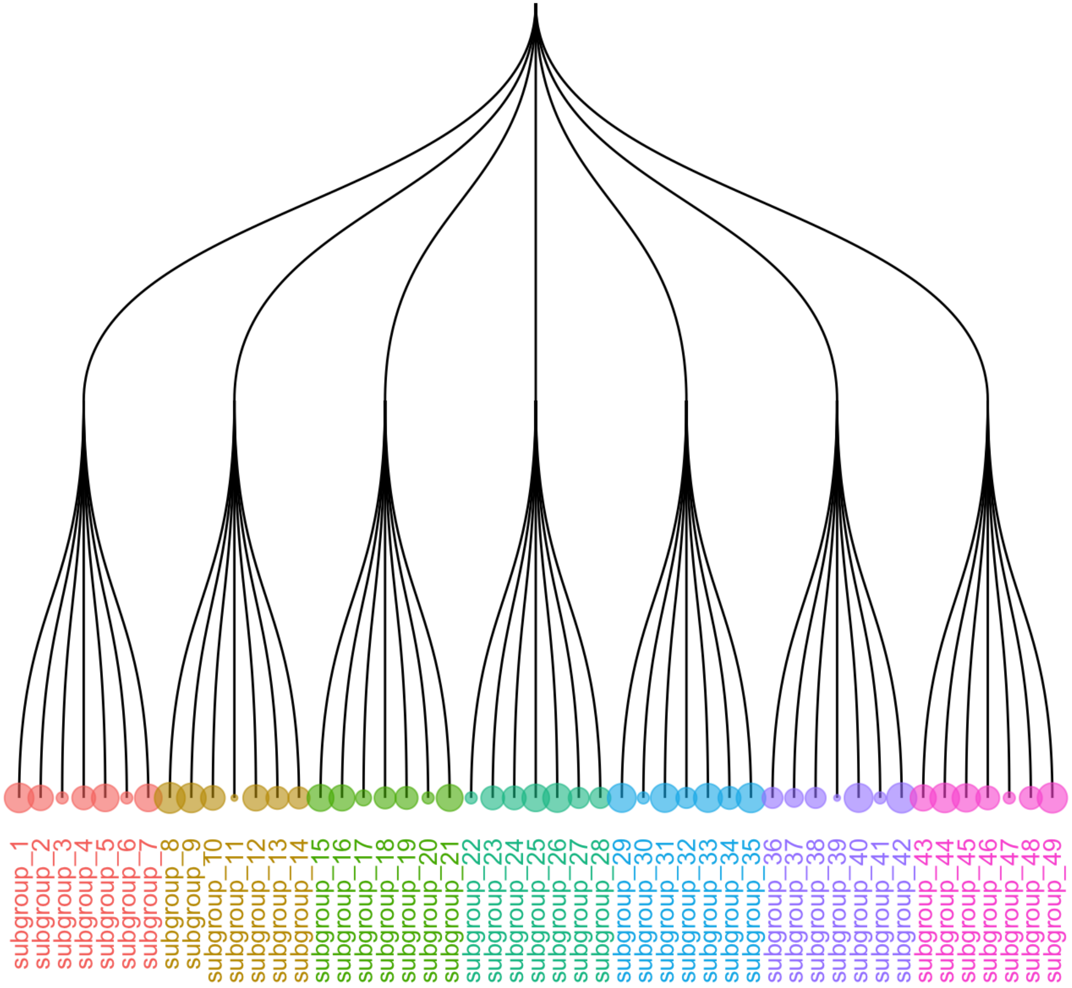
Terminado o processo, teremos um intervalo [a, b] que contém a raiz (e tal que $(b - a) < \varepsilon$) e uma aproximação $\bar{x}$ para a raiz exata.

```r
f <- function(x){
   x-1
}

bisection <- function(f,a,b,tol){
  x <- (a+b)/2
  if ((b-a)<tol) {
    print(x)
    return
    }
  k <- 1
  maxiter <- 20
  while (k < maxiter){
    M <- f(a)
    x <- (a+b)/2
    print(k)
    print(x)
    print(f(x))
    if (M*f(x)>0) {a <- x} else {b <- x}
    k <- k + 1
    if ((b-a)<tol) {return}
  }
}

a <- 0
b <-  10
tol <- 0.1
bisection(f,a,b,tol)
```

```
[1] 18
[1] 1.000023
[1] 2.288818e-05
[1] 19
[1] 1.000004
[1] 3.814697e-06
>
```



https://www.r-graph-gallery.com/335-custom-ggraph-dendrogram.html