UNIVERSIDADE ESTADUAL VALE DO ACARAÚ

CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS

CAMPUS DA CIDAO

CURSO DE MATEMÁTICA

CÁLCULO NUMÉRICO COMPUTACIONAL

FRANCISCO FAGNER PORTELA AGUIAR

LISTA 03 – RESOLUÇÕES

QUESTÃO 01

a)Como temos os comandos ***x:=a*** e ***while(x < b)***, o programa vai verificar todos os valores de ***x*** desde o ***a*** até ***b***, logo, até o intervalo ***[b – delta, b]*** será verificado.

VERDADEIRA

b) O comando ***If (f(x+delta)*f(x) <= 0*** me garante que se a função trocar de sinal (para que ***f(x + delta)*** e ***f(x)*** tenham sinais diferentes) o comando seguinte será executado. O referido comando é ***Then writeln('[',x:2:2,' , ',x+delta:2:2, '] --> f(',x:2:2,')=',f(x):3:4);***

Daí, concluímos que a função irá escrever os valores de ***x***, ***x + delta*** e ***f(x)***. Então, no intervalo ***[x, x + delta]*** estará a raiz, intervalo este de comprimento ***delta***.

VERDADEIRA

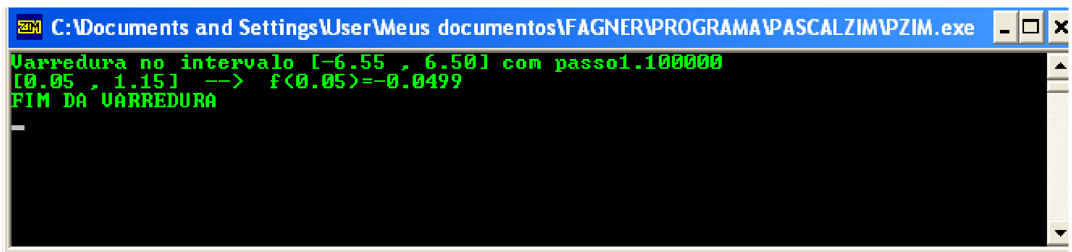
c) Como mostramos na opção anterior e podemos verificar rodando o programa, o intervalo onde está a raiz é impresso sempre, isso é algo certo. A afirmação deste item fala que é possível, e isso dá uma idéia de dúvida e não há dúvida disso.

FALSA

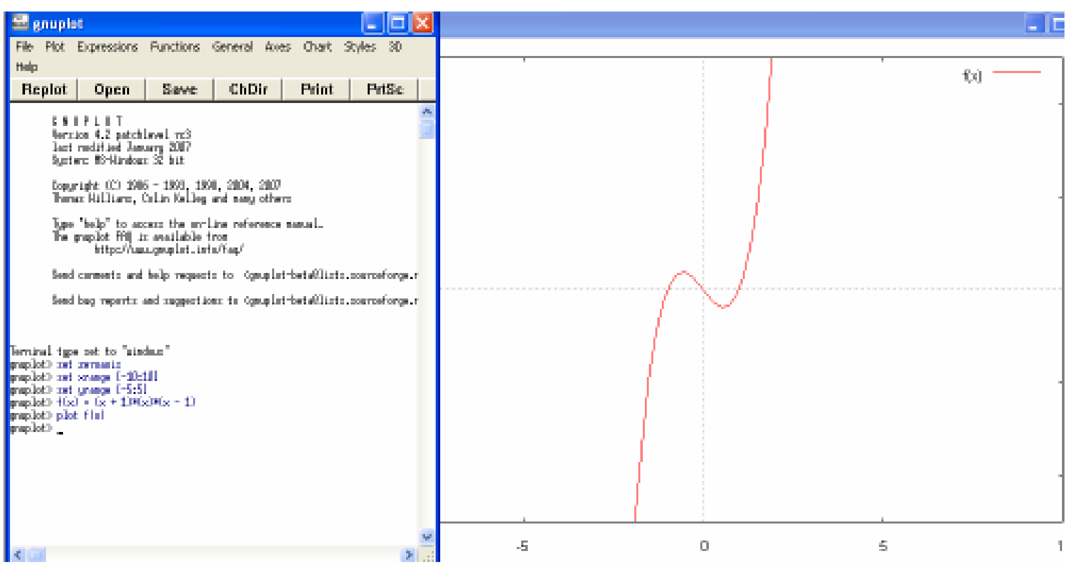
d) Sabendo que uma função ***f*** possui raiz em determinado intervalo ***[a,b]*** podemos escolher um valor de delta para que possamos estudar onde está a raiz. É importante notar que, quanto menor for o valor de delta, mais fácil é de encontrar o ponto onde há troca de sinal. Se tomarmos um intervalo delta um pouco maior teremos a desvantagem de o programa não achar a troca de sinal. Para ilustrar melhor essa situação, fiz uma mudança no programa raizes03_ pas. Atribuí valor 1.1 ao delta e fiz ***f:= (x + 1)*(x)*(x – 1)***. O resultado está abaixo.

```
Program Raizes02_adapt;
{Uses Crt;}
Const
        a = -6.55;
        b = 6.5;
        delta = 1.1;
Var
         x : Real;
Function f(x: Real): Real;
Begin
         f := (x + 1)*(x)*(x - 1);
End;
```

```
Begin
{ClrScr;}
      Writeln('Varredura no intervalo [',a:2:2,' , ',b:2:2,'] com passo',delta);
      x := a;
      While x < b do
      Begin
          If ( f(x+delta)*f(x) <= 0 )
          Then writeln('[',x:2:2,' , ',x+delta:2:2,'] --> f(',x:2:2,')=',f(x):3:4);
          x := x + delta;
      End;
   Writeln('FIM DA VARREDURA');
  readLn;
End.
```



Observando o gráfico abaixo confeccionado no gnuplot, percebemos que há três raízes. Porém, o programa só achou uma pois o valor de delta foi muito grande, então o programa deixou de achar as outras duas raízes.



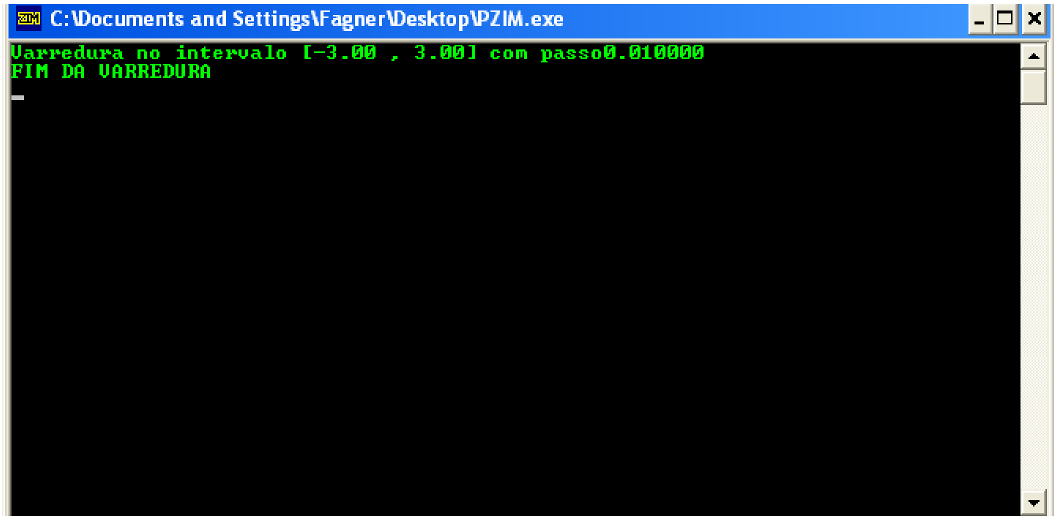
Logo, existe pelo menos um valor de delta que possa dar um intervalo onde haja uma mudança de sinal e, consequentemente, esteja uma raiz da função.

VERDADEIRA

e) Acompanhemos o programa a seguir, que é uma adaptação do programa `raizes03.pas` para o que defende o enunciado deste item, e verifiquemos o que ocorreu.

```
Program Raizes02_adapt;
{Uses Crt;}
Const
    a    = -3.0;
    b    = 3.0;
    delta = 0.01;
Var
   x : Real;
Function f(x: Real): Real;
Begin
    f := x*x;
End;
Begin
    {ClrScr;}
    Writeln('Varredura no intervalo [',a:2:2,' , ',b:2:2,'] com passo',delta);
    x :=  a;
    While  x  <  b  do
    Begin
        If ( f(x+delta)*f(x) <= 0 )
          Then writeln('[',x:2:2,' , ',x+delta:2:2,']  -->  f(',x:2:2,')=',f(x):3:4);
        x :=  x + delta;
    End;
    Writeln('FIM DA VARREDURA');
    readLn;
End.
```

Quando rodamos o programa com essa seqüência de comandos, ele imprime a tela a seguir:



Como podemos perceber, o programa não achou a raiz 0(zero) que a função tem neste intervalo. Isso ocorre porque f é sempre positiva ou zero, assim, jamais o produto de dois valores de f será negativo. Dessa forma, o programa só mostraria onde ***f(x + delta)*f(x)*** fosse 0(zero). Porém, para isso, precisaríamos de um valor de delta muito pequeno. Logo, para uma função que não muda de sinal, o programa pode ou não achar a raiz.

VERDADEIRA

f) Como a função não troca de sinal, ***f(x + delta)*f(x)*** sempre será positivo ou zero, jamais negativo. Portanto, o programa não achará a raiz da função.

FALSA

QUESTÃO 02

a) Suponhamos que ***f*** seja diferenciável. A reta tangente a ***f*** em um ponto ***(a, f(a))*** tem equação dada por

$$y = f(a) + f'(a)(x - a)$$

Desta forma, para tirarmos a raiz da equação da reta devemos ter $y = 0$. Logo:

$$f(a) + f'(a)(x - a) = 0$$

$$f'(a)(x - a) = - f(a)$$

$$x - a = \frac{-f(a)}{f'(a)}$$

$$x = a + \frac{-f(a)}{f'(a)}$$

$$\boldsymbol{x} = \boldsymbol{a} - \frac{f(a)}{f'(a)}$$

Assim, fazendo $\boldsymbol{x = x_0}$ para $\boldsymbol{y = 0}$, temos:

$$\boldsymbol{x_0} = \boldsymbol{a} - \frac{f(a)}{f'(a)}$$

Logo, a sentença é **FALSA.**

b) Pelo que desenvolvemos em **a),** sabemos que

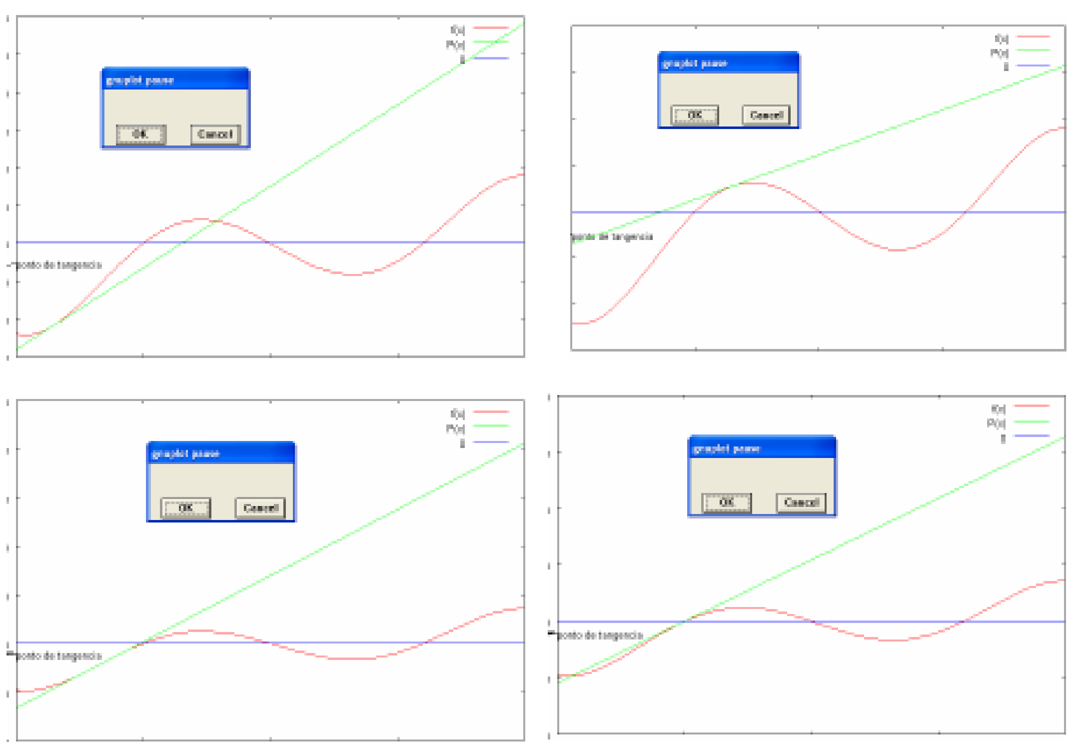
$$\boldsymbol{x_0} = \boldsymbol{a} - \frac{f(a)}{f'(a)}$$

Para isso, devemos ter $\boldsymbol{f'(a) \neq 0}$, pois uma fração não pode ter denominador zero. Isso faz sentido para o nosso caso, pois a função muda de sinal, e isso implica que $\boldsymbol{f'(a) \neq 0}$.

VERDADEIRA

c) FALSA. Para encontrarmos o zero de uma função pelo método da tangente precisamos de que a função mude de sinal para, dessa forma, termos como observar de qual ponto a tangente está se aproximando.

d) Pelas figuras, podemos observar a reta tangente ao gráfico de $\boldsymbol{f}$ em alguns pontos. Esta reta tem coeficiente angular diferente de zero, portanto $\boldsymbol{f}$ é diferenciável. Além disso, como $\boldsymbol{f}$ muda de sinal, a cada vez que tomamos a raiz da reta tangente e traçamos outra tangente, nos aproximamos da raiz de $\boldsymbol{f}$. Portanto, podemos obter uma aproximação da raiz de $\boldsymbol{f}$ usando o zero da reta tangente no ponto $\boldsymbol{(a,f(a))}$.



VERDADEIRA

e) O zero da função que se encontra é por aproximação. Portanto, não é exato. Assim sendo, toda vez que usamos o método da tangente, obtemos um valor que apenas converge para um determinado valor exato.

VERDADEIRA

f) Já sabemos que se ***f*** muda de sinal e é diferenciável podemos encontrar sua raiz aproximada pelo método da tangente. Pela figura, vemos que a função representada possui estas duas características. Dessa forma,podemos encontrar a raiz aproximada dessa função, a qual jamais será exata pois entre dois valores quaisquer podemos tomar infinitos valores, nunca chegando à exatidão.

VERDADEIRA

QUESTÃO 03

a) O programa encontra a raiz aproximada de ***f*** pelo método da secante, portanto, não é a raiz exata. Dessa forma, não podemos dizer que ele encontra o zero.

FALSA

b) No programa podemos localizar o trecho

busca(x,delta)=f(x+delta)*f(x)<=0? x:busca(x+delta, delta)

Aqui podemos observar que a variável busca serve para localizar o extremo do intervalo em que ***f(x + delta)*f(x) <= 0***, ou seja, em que a função tem sinal diferente e, assim, muda de sinal.

VERDADEIRA

c) Pelo que vamos mostrar a seguir, podemos concluir que é **FALSA**.

d) No programa `exer03_04.gnuplot` podemos encontrar o seguinte trecho

P(x) = f(a) + m*(x-a)

em que ***m = (f(b)-f(a))/(b-a)***.

Do valor de ***m*** podemos concluir que o coeficiente angular de ***P(x)*** é o mesmo de uma reta secante a ***f*** nos pontos ***(a, f(a))*** e ***(b, f(b))*** (Pois ***m = (f(b) – f(a))/(b – a)***). Dessa forma, o polinômio ***P*** é a equação da reta secante ao gráfico de ***f*** no intervalo ***[a, b]***.

VERDADEIRA

e) Pelo que podemos depreender da explicação da opção a seguir, percebemos que o programa não procura um único sub-intervalo.

FALSA

f) Pela passagem **printf("Forneça-me o intervalo [a,b] para busca de raizes: \n");** percebemos que o programa pede primeiro o intervalo em que deve ser buscada a mudança de sinal. Observe agora as seguintes passagens:

```
if ( f(a)*f(a+delta)<=0 )
if ( f(a) == 0 )
          {
             printf("Uma raíz exata %f \n", a);
else if ( f(a+delta) == 0 )
           {
               printf("Uma raíz exata %f \n", a+delta);
else
           {
                  printf("Troca de sinal da função no intervalo ");
printf("Raiz p/ método da secante x = %f \n", raiz);
                  printf("Valor de f na raíz aprox f(%f)=%f\n",raiz,f(raiz));
```

A partir delas, podemos concluir que o programa procura todos os sub-intervalos em que haja troca de sinal ou com imagem zero (por causa do **if (f(a)*f(a+delta)<=0**)), aproximando a raiz pelo método da secante, como podemos depreender do comando **printf("Raiz p/ método da secante x = %f \n", raiz);**.

VERDADEIRA

AVALIAÇÃO DO TRABALHO

Saber uma forma de encontrar a raiz de uma função, embora que seja aproximada, é de uma valia muito enorme. Fiquei bastante curioso quando entendi para que serviam os programas. Gostaria de parabenizar a sua atitude em nos ensinar C++. Da forma como estava, apenas os programas vindo em C++ e nós tendo que nos “rebolar” para fazer a lista estava sendo cansativo e sem proveito. Inclusive, na lista anterior que agora peço que desconsidere, tracei várias críticas a isso. É importante aprendermos essa linguagem de programação, e você está contribuindo para isso. As aulas de C++ estão sendo muito proveitosas. Eu gostaria que você discutisse detalhadamente cada questão para, assim, eu aprender mais sobre o assunto. Vou lhe sugerir que cada lista começasse a ser discutida na quinta-feira para ser entregue na quinta-feira seguinte, dessa forma nós teríamos o final de semana para fazer a lista, já tendo discutada-a em sala na aula anterior. De todo o modo, está sendo muito bom fazer essa disciplina, pois estou aprendendo várias coisas interessantes.