Erro no contradomínio: $f(x_f) pprox 0$ Usaremos o erro no domínio, pois vamos encontrar x tal que $f(x)=0$, sendo este x o valor que querer Para realizarmos nossa aproximação, devemos determinar inicialmente uma função sobre a qual aplicare com isto, podemos fazer a aproximação do zero da função $f(x)$, que é o $\sqrt{10}$. Podemos então aplicar o	remos o Método da Bissecção. Para isto, podemos procurar uma função tal que $f(\sqrt{10})=0$. Trivialmente, podemos perceber que pode ser utilizada a função $f(x)=0$
<pre>using LinearAlgebra # Função auxiliar para verificar sinais opostos function tem_sinais_opostos(f,a,b) #função recebe f(a) e f(b) não nulos return f(a)*f(b) < 0 end</pre>	alletamente o algoritmo para lazer uma aproximação.
tem_sinais_opostos (generic function with 1 method) # Método da Bissecção para aproximar raiz de 10 usando o polinômio x²-10 no inter # Como os intervalos não são raízes e já tem sinais opostos, não nos preocuparemo function bissecao_raiz_10(erro) # Intervalo [0,20]	
<pre>a = 0 b = 20 f(x) = x^2-10 # Função x²-10 tamanho_final_do_intervalo = 2*erro iteracoes = floor(log2((b-a)/tamanho_final_do_intervalo))+1 for i=1:iteracoes m=(a+b)/2 if f(m) == 0 return m end if tem_sinais_opostos(f,a,m) b=m else a=m</pre>	
<pre>end end x_final=(a+b)/2 return x_final, iteracoes end bissecao_raiz_10 (generic function with 1 method)</pre>	
Com isto, podemos fazer a chamada do método passando o tamanho máximo do erro que o usuário solid bissecao_raiz_10(10^-8) $(3.162277666851878,\ 30.0)$ Com isto, obtemos uma aproximação de $\sqrt{10}\approx 3.162277666851878$ onde foram necessários 30 pass	icita, pois o tamanho do intervalo final define uma distância ao x real. Com isto, podemos obter o número de iterações necessárias para que o erro seja, no máximo, 1 sos para obter um erro máximo de 10^{-8} .
	os indicar se as funções na bissecção tem sinais opostos. O segundo método é uma implementação genérica do Método de Newton, que receberá uma função, a deri o do valor. O terceiro método será o Método da Bissecção, que receberá como parâmetro o polinômio de grau 5 e sua derivada. Esta última função realizará normalme létodo de Newton passando como chute inicial essa aproximação e retornará esse valor final.
<pre># Função auxiliar para verificar sinais opostos function tem_sinais_opostos(f,a,b) #função recebe f(a) e f(b) não nulos return f(a)*f(b) < 0 end tem_sinais_opostos (generic function with 1 method) # Método de Newton para a função f e f'</pre>	
<pre># Método de Newton para a função f e f' function metodo_newton(f, fd, iteracoes, chute) # Função f, f', número de iteraco for i=1:iteracoes</pre>	
<pre># Método da Bissecção para retornar uma raíz de um polinômio de grau 5 usando o M function bissecao(f, fd) # Intervalo [-100, 100] a = -100 b = 100 tamanho_final_do_intervalo = 10^-2</pre>	Método da Bissecção e o Método de Newton.
<pre>iteracoes = floor(log2((b-a)/tamanho_final_do_intervalo))+1 if !(tem_sinais_opostos(f,a,b)) # Verificação de sinais opostos return "Não tem sinais opostos" end for i=1:iteracoes m=(a+b)/2 if f(m) == 0 return m end if tem_sinais_opostos(f,a,m) b=m</pre>	
<pre>else a=m end end chute=(a+b)/2 x_final = metodo_newton(f, fd, 20, chute) return x_final</pre>	
bissecao (generic function with 1 method) Com isto, realizaremos uso do Método da Bissecção em conjunto com o Método de Newton para calcular $f(x) = x^5-25$ $fd(x) = 5x^4$	ur uma das raízes de $f(x)=x^5-25$. Este método deve então nos dar uma aproximação de $\sqrt[5]{25}$.
bissecao(f, fd) 1.9036539387158784 Exercício 1.3)	saremos o Método da Bissecção para realizar o cálculo. Devemos garantir também um intervalo menor que 10^{-3}
Para obtermos a aproximação, podemos utilizar uma aproximação pelo domínio, procurando alguma funç p rocuraremos no domínio da função o valor da raíz $ln(3)$, obtendo uma aproximação desta. Para isto, construiremos primeiro o método auxiliar de sinais apostos. Após insto, faremos o Método da E	ição tal que $f(ln(3))=0$. Para isto, como sabemos calcular e^x para qualquer x e sabemos as propriedades de ln , podemos utilizar a função $f(x)=3-e^x$, pois a
<pre># Função auxiliar para verificar sinais opostos function tem_sinais_opostos(f,a,b) #função recebe f(a) e f(b) não nulos return f(a)*f(b) < 0 end tem_sinais_opostos (generic function with 1 method) # Método da Bissecção para aproximar ln(3) usando a função 3 - exp(x) no interval</pre>	lo [-5,5]
<pre># Metodo da Bissecçao para aproximar th(s) dsando a runção 3 - exp(x) no intervato # Como os intervalos não são raízes e já tem sinais opostos, não nos preocuparemo function bissecao_ln_3(tamanho_final_do_intervalo) # Intervalo [-5,5] a = -5 b = 5 f(x) = 3 - exp(x) # Função 3-exp(x) iteracoes = floor(log2((b-a)/tamanho_final_do_intervalo))+1</pre>	
<pre>iteracoes = floor(log2((b-a)/tamanho_final_do_intervalo))+1 for i=1:iteracoes m=(a+b)/2 if f(m) == 0 return m end if tem_sinais_opostos(f,a,m)</pre>	
<pre>b=m else a=m end end x_final=(a+b)/2</pre>	
return x_final end bissecao_ln_3 (generic function with 1 method) Com isto, podemos utilizar a função do Método da Bissecção criado para este caso passando um tamanh bissecao_ln_3(10^-4)	ho de intervalo menor que 10^{-3} para nos dar uma aproximação de $ln(3)$
1.0985946655273438 Exercício 1.4)	ação polinomial. Utilizaremos então um conjunto de pontos afim de montar o sistema de equações que resultará na matriz de Vandermonde para encontrarmos uma fu
/amos utilizar apenas os valores entre cos(30) e cos(60), pois assim podemos tirar uma aproximação da pontos $(x_1,y_1),(x_2,y_2),(x_3,y_3)$ e que nos dará um polinômio na forma	a função trigonométrica próxima de um polinômio de grau 2 utilizando valores já conhecidos, isto é, cos(30), cos(45) e cos(60). Com isto, teremos uma função que pas $P=c_2x^2+c_1x+c_0$
Com isto, podemos montar o seguinte sistema de equações	$egin{cases} c_2x_0^2+c_1x_0+c_0=y_0\ c_2x_1^2+c_1x_1+c_0=y_1\ c_2x_2^2+c_1x_2+c_0=y_2 \end{cases}$
Jsaremos então os pontos $(30,\frac{\sqrt{3}}{2}),(45,\frac{\sqrt{2}}{2}),(60,\frac{1}{2}).$ Portanto, temos que	$egin{cases} c_2 30^2 + c_1 30 + c_0 &= rac{\sqrt{3}}{2} \ c_2 45^2 + c_1 45 + c_0 &= rac{\sqrt{2}}{2} \ c_2 60^2 + c_1 60 + c_0 &= rac{1}{2} \end{cases}$
Com isto, podemos modelar as matrizes da seguinte forma	$egin{align} \left(c_2 60^2 + c_1 60 + c_0 = rac{1}{2} ight. \ V = egin{bmatrix} 1 & 30 & 30^2 \ 1 & 45 & 45^2 \end{bmatrix} \end{split}$
	$y = \begin{bmatrix} \frac{\sqrt{3}}{2} \\ \frac{\sqrt{2}}{2} \\ \frac{1}{2} \end{bmatrix}$
	$\left[egin{array}{c} rac{1}{2} \ \end{array} ight] \ C = \left[egin{array}{c} c_0 \ c_1 \ c_2 \ \end{array} ight]$
Devemos então resolver o sistema tal que $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cossero $V \cdot c = y$. Vamos então utilizar o método interpolação_cos $V \cdot c = y$. Vamos então utilizar o método interpolação_cos $V \cdot c = y$. Vamos então utilizar o método interpolação utilizar o mé	no_40 descrito abaixo para resolver o sistema, retornando os coeficientes do polinômio de grau 2.
<pre>c=V\y # Resolve o sistema linear Vc=y return c #vetor de coeficientes end interpolação_cosseno (generic function with 1 method) com isto, podemos utilizar o método para achar os coeficientes da polinômio de grau 2 que nos dá uma a coefs = interpolação_cosseno()</pre>	aproximação da função original. Podemos então armazenar esses coeficientes, montar a função quadrática e calcular uma aproximação de cos(40) a partir desta funç
c2 = coefs[3] c1 = coefs[2] c0 = coefs[1] f(x) = c2*(x^2) + c1*x + c0 f(deg2rad(40)) 0.7654338952290287 Exercício 1.5) Para este exercício, devemos descobrir o horário ou uma interpolação do horário em que houve um assa 2.5h00: 34°C	assinato. De acordo com a descrição, as tempratura registradas foram
.6h30: 30°C .7h30: 25°C Itilizando interpolação polinomial, podemos gerar uma aproximação de uma função que represente o ca na forma	aso descrito utilizando os pontos $(15,34),(16.5,30),(17.5,25)$, pois correlacionam cada hora com uma temperatura registrada. Com isto, temos vamos ter na saída 0
Podemos então montar o seguinte sistema de equações Com isto, podemos montar o seguinte sistema de equações	$P = c_2 x^2 + c_1 x + c_0$
Com isto, podemos modelar as matrizes da seguinte forma	$egin{aligned} & \left\{ egin{aligned} c_2 x_0^2 + c_1 x_0 + c_0 &= y_0 \ c_2 x_1^2 + c_1 x_1 + c_0 &= y_1 \ c_2 x_2^2 + c_1 x_2 + c_0 &= y_2 \end{aligned} ight. \ & V = egin{bmatrix} 1 & 15 & 15^2 \ 1 & 16.5 & 16.5^2 \end{bmatrix} \end{aligned}$
	$V = egin{bmatrix} 1 & 16.5 & 16.5^2 \ 1 & 17.5 & 17.5^2 \end{bmatrix} \ y = egin{bmatrix} 34 \ 30 \ 25 \end{bmatrix}$
Devemos então resolver o sistema tal que $V\cdot c=y$. Vamos então utilizar o método interpolacao_crime c	$C=egin{bmatrix} c_0 \ c_1 \ c_2 \end{bmatrix}$ descrito abaixo para resolver o sistema, retornando os coeficientes do polinômio de grau 2.
<pre>function interpolacao_crime() # Cria a matriz V x = [15; 16.5; 17.5] y = [34; 30; 25] V=[x.^0 x.^1 x.^2] c=V\y # Resolve o sistema linear Vc=y return c #vetor de coeficientes end</pre>	
<pre>interpolacao_crime (generic function with 1 method) com isto, podemos gerar uma função polinomial e um tempo aproximado do horário da morte. using Plots coefs = interpolacao_crime() c2 = coefs[3] c1 = coefs[2]</pre>	
$c0 = coefs[1]$ $f(x) = c2*(x^2) + c1*x + c0$ $plot(f, 10, 20)$ 35	
25	
15 -	
	ecidos, nos indicando a taxa de variação com que a temperatura cai. Com isto, podemos ter uma boa aproximação do horário de morte se acharmos o ponto máximo d
Para isso, podemos utilizar o método da bissecção para acharmos a raíz da derivada da função gerada. # Função auxiliar para verificar sinais opostos function tem_sinais_opostos(f,a,b) #função recebe f(a) e f(b) não nulos return f(a)*f(b) < 0 end # Mótodo da Bissocsão para aproximar a raiz da função derivada da função interpol	
<pre># Método da Bissecção para aproximar a raiz da função derivada da função interpol function bisseccao_crime(tamanho_final_do_intervalo, c1,c2) # Intervalo [15,40] a = 0 b = 40 f(x) = (2*c2)*x + c1 iteracoes = floor(log2((b-a)/tamanho_final_do_intervalo))+1</pre>	
<pre>for i=1:iteracoes m=(a+b)/2 if f(m) == 0 return m end if tem_sinais_opostos(f,a,m) b=m</pre>	
<pre>b=m else a=m end end x_final=(a+b)/2 return x_final</pre>	
<pre>end bisseccao_crime (generic function with 2 methods) Com isto, podemos procurar na derivada da função quadrática encontrada sua raiz, indicando o ponto en using Plots coefs = interpolacao_crime()</pre>	n que a temperatura do corpo começou a decair, sendo o horário mais provável do seu assassinato.
c2 = coefs[3] c1 = coefs[2] c0 = coefs[1] bisseccao_crime(10^8 , c1, c2) 14.321428569965065 Com isto, temos que o provável horário do assassinato foi $14h18\sim14h20$.	
Exercício 1.6) Exercício 1.7)	
Para este exercício, devemos realizar uma interpolação por partes dados 5 pontos $\{(x_0,y_0),(x_1,y_1),(x_1,y_2),(x_2,y_3)\}$ Com isto, temos dois polinômios, P e Q, na forma	$(x_2,y_2),(x_3,y_3),(x_4,y_4)$ }, produzindo na saída um polinômio de grau 3 sem bicos (definição). $P=c_3x^3+c_2x^2+c_1x+c_0$ $Q=d_3x^3+d_2x^2+d_1x+d_0$
ue devem ser interpolados. Podemos então montar inicialmente o seguinte sistema de equações	$\left(egin{array}{l} P(x_0) = y_0 \ P(x_1) = y_1 \end{array} ight)$
	$\begin{cases} P(x_2)=y_2 \ Q(x_2)=y_2 \ Q(x_3)=y_3 \ Q(x_4)=y_4 \end{cases}$ troduzir mais 2 equações para que o sistema possa ser resolvido. Vamos introduzir 2 variáveis: uma relativa à derivada de P e Q em x_2 , para que o ponto de encontro
	ma curvatura em x_2 . $ \int P(x_0) = y_0$
Como este sistema possui 8 variáveis $(c_0,c_1,c_2,c_3,d_0,d_1,d_2,d_3,)$ e apenas 6 equações, devemos intenham a mesma inclinação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm ${ m Com}$ isto, temos que	$egin{aligned} P(x_1) &= y_1 \ P(x_2) &= y_2 \end{aligned}$
enham a mesma inclinação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm Com isto, temos que	$\left\{egin{aligned} P(x_2) &= y_2 \ Q(x_2) &= y_3 \ Q(x_4) &= y_4 \ P'(x_2) &= Q'(x_2) \ P''(x_2) &= Q''(x_2) \end{aligned} ight.$
enham a mesma inclinação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm	$\begin{cases} P(x_2) = y_2 \\ Q(x_2) = y_2 \\ Q(x_3) = y_3 \\ Q(x_4) = y_4 \\ P'(x_2) = Q'(x_2) \\ P''(x_2) = Q''(x_2) \end{cases}$ ma $\begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 & 0 & 0 & 0 & 0 \\ 1 & x_1 & x_1^2 & x_1^3 & 0 & 0 & 0 & 0 \\ 1 & x_2 & x_2^2 & x_2^3 & 0 & 0 & 0 & 0 \end{bmatrix}$
enham a mesma inclinação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm Com isto, temos que	$V = \begin{cases} P(x_2) = y_2 \\ Q(x_2) = y_2 \\ Q(x_3) = y_3 \\ Q(x_4) = y_4 \\ P'(x_2) = Q'(x_2) \end{cases}$ $P''(x_2) = Q''(x_2)$ $P''(x_2) = Q''(x_2)$ $P''(x_2) = Q''(x_2)$ $Q''(x_2) = Q''(x_2)$
enham a mesma inclinação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm Com isto, temos que	$V = \begin{cases} P(x_2) = y_2 \\ Q(x_2) = y_2 \\ Q(x_3) = y_3 \\ Q(x_4) = y_4 \\ P'(x_2) = Q'(x_2) \\ P''(x_2) = Q''(x_2) \end{cases}$ ma $V = \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 & 0 & 0 & 0 & 0 \\ 1 & x_1 & x_1^2 & x_1^3 & 0 & 0 & 0 & 0 \\ 1 & x_2 & x_2^2 & x_2^3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & x_2 & x_2^2 & x_2^2 \\ 0 & 0 & 0 & 0 & 1 & x_3 & x_3^3 & x_3^3 \\ 0 & 0 & 0 & 0 & 1 & x_4 & x_4^2 & x_4^3 \\ 0 & 1 & 2x_2 & 3x_2^2 & 0 & -1 & -2x_2 & -3x_2^2 \end{cases}$
enham a mesma inclinação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm Com isto, temos que	$\begin{cases} P(x_2) = y_2 \\ Q(x_2) = y_3 \\ Q(x_4) = y_4 \\ P'(x_2) = Q'(x_2) \end{cases}$ The second of the proof of
enham a mesma inclinação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm Com isto, temos que	$V = \begin{bmatrix} P(x_2) = y_2 \\ Q(x_2) = y_2 \\ Q(x_3) = y_3 \\ Q(x_4) = y_4 \\ P'(x_2) = Q'(x_2) \\ P''(x_2) = Q''(x_2) \end{bmatrix}$ ma $V = \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 & 0 & 0 & 0 & 0 \\ 1 & x_1 & x_1^2 & x_1^3 & 0 & 0 & 0 & 0 \\ 1 & x_2 & x_2^2 & x_2^3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & x_2 & x_2^2 & x_2^3 \\ 0 & 0 & 0 & 0 & 1 & x_3 & x_3^2 & x_3^3 \\ 0 & 0 & 0 & 0 & 1 & x_4 & x_4^2 & x_4^3 \\ 0 & 1 & 2x_2 & 3x_2^2 & 0 & -1 & -2x_2 & -3x_2^2 \\ 0 & 0 & 2 & 6x_2 & 0 & 0 & -2 & -6x_2 \end{bmatrix}$ $y = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_2 \\ y_3 \\ y_4 \\ 0 \\ 0 \end{bmatrix}$
### Function interpolacao_por_partes_cubica(x::Array(Int64),y::Array(Int64)) #### Cria a matriz V V = [1 x[1] x[1]^2 x[1]^3 0 0 0; 1 x[2] x[2]^2 x[2]^3 0 0 0; 1 x[3] x[3]^2 y = [V]; y # Resolve o sistema linear Vc=y C=Vy # Resolve o sistema linear Vc=y	$\begin{cases} P(x_2) = y_2 \\ Q(x_2) = y_2 \\ Q(x_3) = y_3 \\ Q(x_4) = y_4 \\ P'(x_2) = Q'(x_2) \end{cases}$ $Q(x_4) = y_4$ $P'(x_2) = Q''(x_2)$ $V = \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 & 0 & 0 & 0 & 0 & 0 \\ 1 & x_1 & x_2^2 & x_3^3 & 0 & 0 & 0 & 0 & 0 \\ 1 & x_2 & x_2^2 & x_2^3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & x_2 & x_2^2 & x_3^3 \\ 0 & 0 & 0 & 0 & 1 & x_4 & x_1^2 & x_1^3 \\ 0 & 1 & 2x_2 & 3x_2^2 & 0 & -1 & -2x_2 & -3x_2^2 \\ 0 & 0 & 2 & 6x_2 & 0 & 0 & 2 & -6x_2 \end{bmatrix}$ $y = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ 0 \\ 0 \end{bmatrix}$
### function interpolacao_por_partes_cubica(x::Array{Int64}, y::Array{Int64}) ### function interpolacao_por_partes_cubica(x::Array{Int64}, y::Array{Int64}) ### Cris a matriz V V = [1 x[1] x[1]^2 x[1]^3 0 0 0 0; 1 x[2] x[2]^2 x[2]^3 0 0 0 0; 1 x[3] x[3]^2 Y = [[9]]; y[2]; y[3]; y[3]; y[4]; y[5]; 0; 0]	$P(x_2) = y_1 \\ Q(x_3) = y_3 \\ Q(x_4) = y_2 \\ Q(x_2) = y_3 \\ Q(x_2) = y_3 \\ Q(x_2) = y_3 \\ Q(x_2) = y_4 \\ P'(x_2) = Q'(x_2) \\ P''(x_2) = Q''(x_2) \\ P''(x_2) = Q''(x_2) \\ P''(x_2) = Q''(x_2) \\ P''(x_2) = Q''(x_2) \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0$
function interpolação, e a segunda relativa a segunda derivada com a finalidade de terem a mesm com isto, temos que function interpolação, por_partes_cubica(x::Array{Int64}, y::Array{Int64}) # Cria a matriz !/ # Cria a matriz !/ # I [12] x [1] 3 0 0 0 0 0; 1 x [2] x [2] 1 x [2] 2 x [2] 3 0 0 0 0; 1 x [3] x [3] 2 y = [y[1]; y[2]; y[3]; y[4]; y[5]; 0; 0] c=Vy # Resolve o sistema linear Vc=y return c # Vetor de coeficientes end interpolação_por_partes_cubica (generic function with 1 method) com os coeficientes, podemos então plotar as funções encontradas respectivamente em [0,2] e [2,4] par.	$P(x_2) = y_1 \\ Q(x_3) = y_3 \\ Q(x_4) = y_2 \\ Q(x_2) = y_3 \\ Q(x_2) = y_3 \\ Q(x_2) = y_3 \\ Q(x_2) = y_4 \\ P'(x_2) = Q'(x_2) \\ P''(x_2) = Q''(x_2) \\ P''(x_2) = Q''(x_2) \\ P''(x_2) = Q''(x_2) \\ P''(x_2) = Q''(x_2) \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0$