

The background of the slide is a dark, textured composition. On the left, there is a portrait of Isaac Newton, showing his characteristic long, wavy hair and a serious expression. Overlaid on and around the portrait are various mathematical elements. To the left of Newton's head, there is a diagram with a circle and a point labeled 'A' and 'B'. Above it, some text in a script font is visible, including 'T. A. R. 3'. To the right of Newton's head, there is a complex geometric diagram with a circle, a point labeled 'F', and a line segment labeled 'P'. Above this diagram, the formula $E = ma$ is written. Below the main title, the word 'QUADRÁTICA' is written in a serif font. At the bottom, the phrase 'OU GEOMÉTRICA' is written in a smaller serif font.

INTERPOLAÇÃO

QUADRÁTICA

OU GEOMÉTRICA

FUNDAMENTO S

01

A interpolação quadrática é uma técnica que nos permite estimar um valor intermediário entre dois pontos conhecidos. Ela utiliza uma função polinomial quadrática para traçar uma curva suave que passa pelos pontos dados. Essa técnica é amplamente utilizada em diversas áreas, como ciência, engenharia e computação, para prever valores desconhecidos ou preencher lacunas em conjuntos de dados.

FUNDAMENTOS

METÓDO

01

A interpolação quadrática é uma técnica que nos permite estimar um valor intermediário entre dois pontos conhecidos. Ela utiliza uma função polinomial quadrática para traçar uma curva suave que passa pelos pontos dados. Essa técnica é amplamente utilizada em diversas áreas, como ciência, engenharia e computação, para prever valores desconhecidos ou preencher lacunas em conjuntos de dados.

02

Imagine que você tenha um conjunto de dados com pontos conhecidos, mas deseje estimar o valor de um ponto intermediário. A interpolação quadrática é uma abordagem eficaz para resolver esse problema. Ela utiliza a forma de uma função polinomial quadrática para ajustar uma curva suave aos pontos dados, permitindo-nos encontrar o valor desejado com base nas informações disponíveis.

FUNDAMENTOS

METÓDO

INTERPOLAÇÃO

01

A interpolação quadrática é uma técnica que nos permite estimar um valor intermediário entre dois pontos conhecidos. Ela utiliza uma função polinomial quadrática para traçar uma curva suave que passa pelos pontos dados. Essa técnica é amplamente utilizada em diversas áreas, como ciência, engenharia e computação, para prever valores desconhecidos ou preencher lacunas em conjuntos de dados.

02

Imagine que você tenha um conjunto de dados com pontos conhecidos, mas deseje estimar o valor de um ponto intermediário. A interpolação quadrática é uma abordagem eficaz para resolver esse problema. Ela utiliza a forma de uma função polinomial quadrática para ajustar uma curva suave aos pontos dados, permitindo-nos encontrar o valor desejado com base nas informações disponíveis.

03

A interpolação quadrática é uma técnica poderosa para a previsão de valores desconhecidos com base em dados limitados. Com apenas três pontos conhecidos, podemos traçar uma curva suave que se ajusta aos pontos e estimar valores intermediários com boa precisão. Essa técnica é especialmente útil quando os dados seguem um padrão não linear e desejamos preencher as informações faltantes.

FUNDAMENTO S

METÓDO

INTERPOLAÇÃO

QUADRÁTICA

$$\begin{cases} a_2x_0^2 + a_1x_0 + a_0 = y_0 \\ a_2x_1^2 + a_1x_1 + a_0 = y_1 \\ a_2x_2^2 + a_1x_2 + a_0 = y_2 \end{cases}$$

01

A interpolação quadrática é uma técnica que nos permite estimar um valor intermediário entre dois pontos conhecidos. Ela utiliza uma função polinomial quadrática para traçar uma curva suave que passa pelos pontos dados. Essa técnica é amplamente utilizada em diversas áreas, como ciência, engenharia e computação, para prever valores desconhecidos ou preencher lacunas em conjuntos de dados.

02

Imagine que você tenha um conjunto de dados com pontos conhecidos, mas deseje estimar o valor de um ponto intermediário. A interpolação quadrática é uma abordagem eficaz para resolver esse problema. Ela utiliza a forma de uma função polinomial quadrática para ajustar uma curva suave aos pontos dados, permitindo-nos encontrar o valor desejado com base nas informações disponíveis.

03

A interpolação quadrática é uma técnica poderosa para a previsão de valores desconhecidos com base em dados limitados. Com apenas três pontos conhecidos, podemos traçar uma curva suave que se ajusta aos pontos e estimar valores intermediários com boa precisão. Essa técnica é especialmente útil quando os dados seguem um padrão não linear e desejamos preencher as informações faltantes.

04

Ela pode ser uma ferramenta valiosa para a análise e interpretação de conjuntos de dados. Com ela, podemos criar curvas suaves que se aproximam dos pontos conhecidos, permitindo-nos extrapolar informações e fazer previsões confiáveis. Seja na modelagem matemática, na análise de dados experimentais ou na construção de gráficos, a interpolação quadrática desempenha um papel fundamental para melhor compreender e utilizar os dados disponíveis.

ESTE CODIGO AO
LADO
GERA UM GRAFICO
QUE REPRESENTA
O METODO DA
INTERPOLAÇÃO
QUADRATICA

```
import numpy as np
import matplotlib.pyplot as plt

def interpolate_quadratic(x, y, t):
    u = 1 - t
    tt = t * t
    uu = u * u

    result = uu * y[0] + 2 * u * t * y[1] + tt * y[2]
    return result

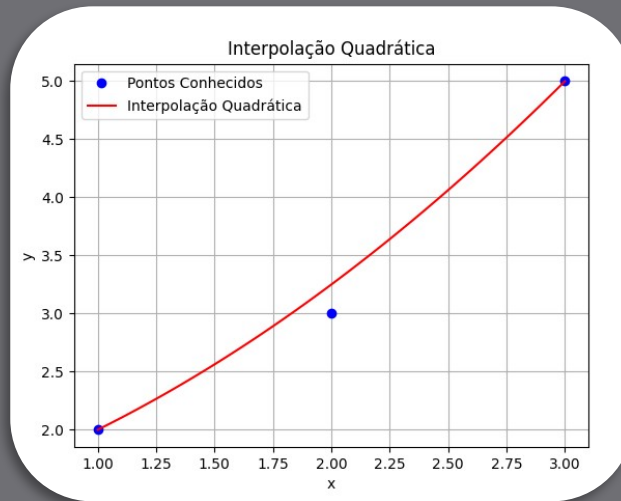
Pontos conhecidos
[1, 2, 3]
[2, 3, 5]

Interpolados
interpolated_x = np.linspace(x[0], x[2], num=100)
interpolated_y = [interpolate_quadratic(x, y, t) for t in np.linspace(0, 1, num=100)]

gráfico
plt.plot(x, y, 'bo', label='Pontos Conhecidos')
plt.plot(interpolated_x, interpolated_y, 'r-', label='Interpolação Quadrática')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Interpolação Quadrática')
plt.legend()
grid(True)
show()
```

SEGUIMOS

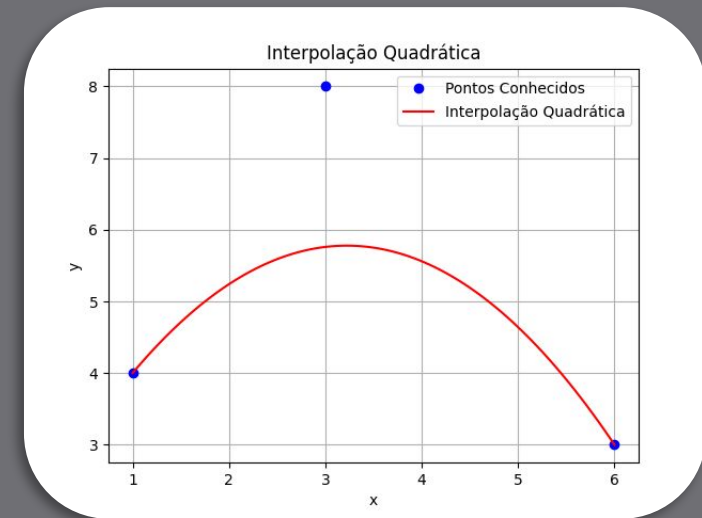
Ao aplicar o método da interpolação quadrática, o gráfico resultante apresenta uma curva suave e contínua, passando pelos pontos conhecidos. Através da interpolação, seriam estimados valores intermediários entre os pontos originais, fornecendo uma representação aproximada da função quadrática subjacente.



O gráfico mostra uma curva ascendente, indicando um crescimento gradual dos valores à medida que x aumenta.

OBS

Este gráfico foi gerado a partir do mesmo código anterior, apenas alterando os valores de x e y.



Dado que a interpolação quadrática envolve uma função quadrática, a curva obtém-se um formato curvilíneo bem profundo.

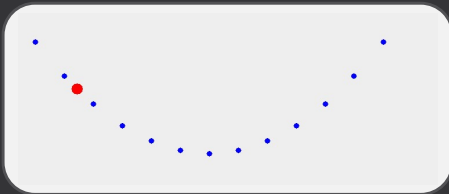


SIMULAÇÃO

EM

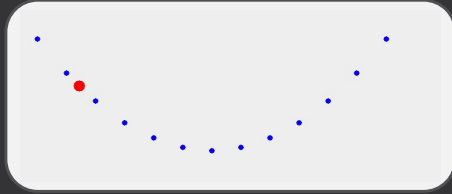
JAVA

ETAPA 1



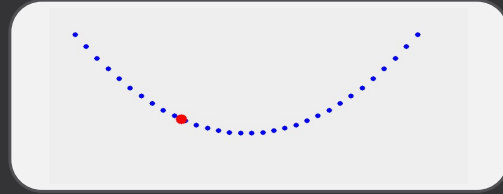
O código realiza uma animação que representa uma curva de interpolação quadrática e um objeto se movendo ao longo dessa curva.

ETAPA 1



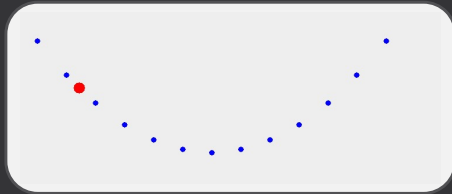
O código realiza uma animação que representa uma curva de interpolação quadrática e um objeto se movendo ao longo dessa curva.

ETAPA 2



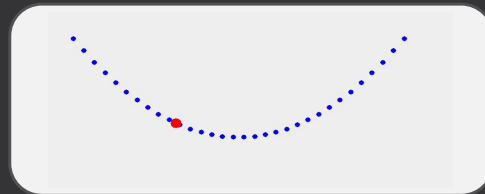
A janela é configurada para exibir a animação e tem um tamanho pré-definido de 800x600 pixels.

ETAPA 1



O código realiza uma animação que representa uma curva de interpolação quadrática e um objeto se movendo ao longo dessa curva.

ETAPA 2



A janela é configurada para exibir a animação e tem um tamanho pré-definido de 800x600 pixels.

ETAPA 3



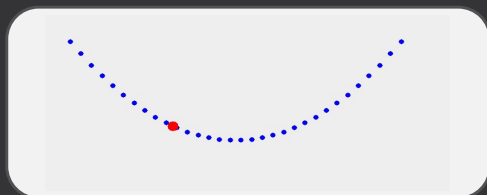
A curva é desenhada usando o método de interpolação quadrática, onde são calculados os pontos intermediários da curva e são representados como círculos azuis.

ETAPA 1



O código realiza uma animação que representa uma curva de interpolação quadrática e um objeto se movendo ao longo dessa curva.

ETAPA 2



A janela é configurada para exibir a animação e tem um tamanho pré-definido de 800x600 pixels.

ETAPA 3

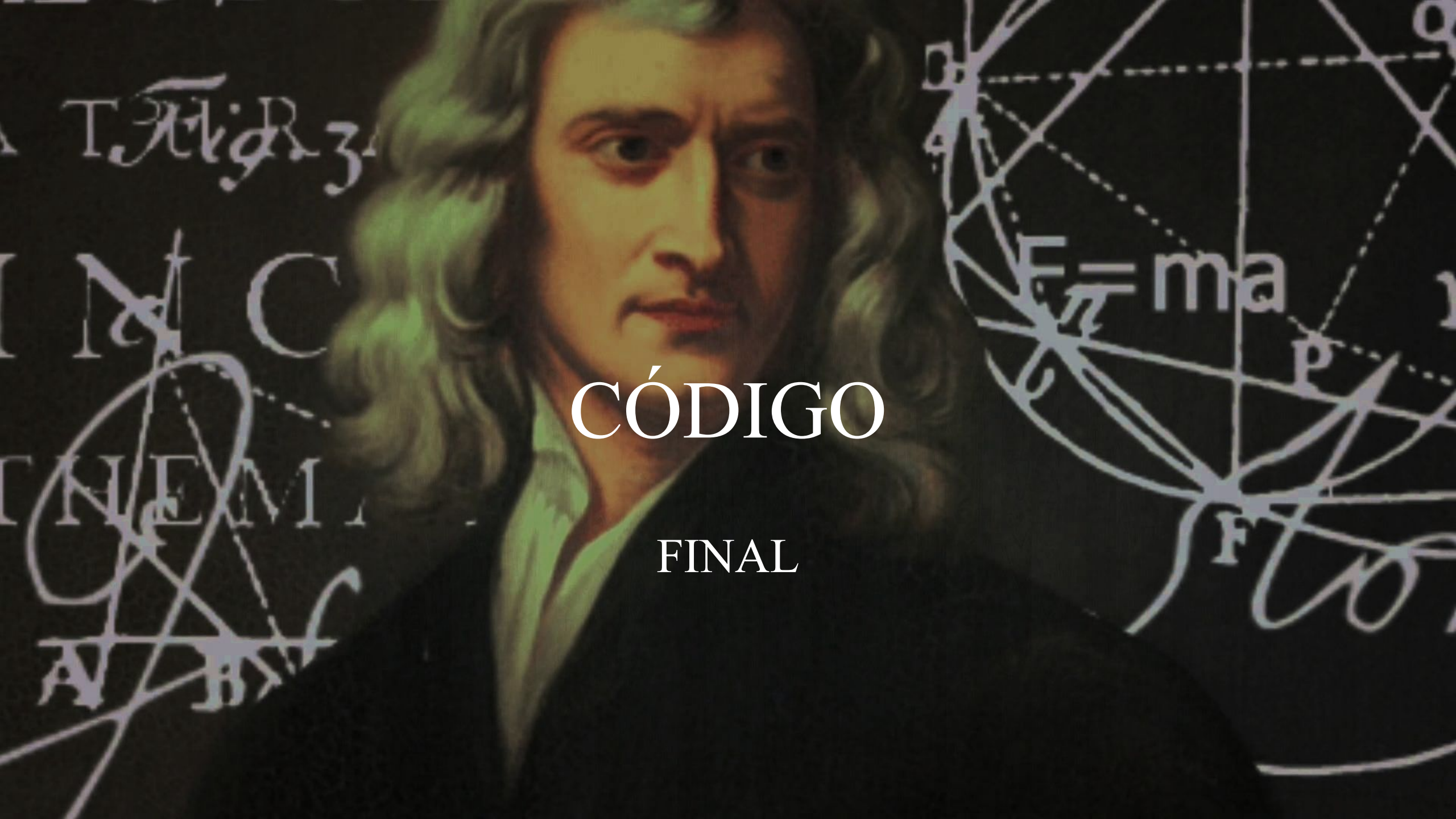


A curva é desenhada usando o método de interpolação quadrática, onde são calculados os pontos intermediários da curva e são representados como círculos azuis.

ETAPA 4



O objeto animado segue a curva de interpolação quadrática, alterando sua posição com o passar do tempo. A posição é calculada com base em um parâmetro que varia entre 0 e 1, representando o progresso da animação.



CÓDIGO

FINAL


```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class AnimacaoInterpolacaoQuadratica extends JFrame implements ActionListener {
    private static final int WIDTH = 800;
    private static final int HEIGHT = 600;

    private Timer timer;
    private int currentStep;

    private int startX = 100;
    private int startY = 100;
    private int controlX = 400;
    private int controlY = 500;
    private int endX = 700;
    private int endY = 100;
    private int objectX;
    private int objectY;

    public AnimacaoInterpolacaoQuadratica() {
        setTitle(title:"Animação com a Interpolação Quadrática");
        setSize(WIDTH, HEIGHT);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        currentStep = 0;

        timer = new Timer(delay:10, this);
        timer.start();
    }

    public void paint(Graphics g) {
        super.paint(g);
        Graphics2D g2d = (Graphics2D) g;

        // Desenha a curva de interpolação quadrática
        g2d.setStroke(new BasicStroke(width:2));
        g2d.setColor(Color.BLUE);
        for (int i = 0; i <= currentStep; i++) {
            double t = (double) i / currentStep;
            int x = interpolacaoQuadratica(startX, controlX, endX, t);
            int y = interpolacaoQuadratica(startY, controlY, endY, t);
            g2d.fillOval(x - 5, y - 5, width:10, height:10);
        }

        // Desenha o objeto animado
        g2d.setColor(Color.RED);
        g2d.fillOval(objectX - 10, objectY - 10, width:20, height:20);
    }
}

```

```
}  
  
public int interpolacaoQuadratica(int p0, int p1, int p2, double t) {  
    double u = 1 - t;  
    double tt = t * t;  
    double uu = u * u;  
  
    return (int) (uu * p0 + 2 * u * t * p1 + tt * p2);  
}
```

```
public void actionPerformed(ActionEvent e) {  
    if (currentStep <= 100) {  
        currentStep++;  
    } else {  
        currentStep = 0;  
    }  
  
    double t = (double) currentStep / 100;  
    objectX = interpolacaoQuadratica(startX, controlX, endX, t);  
    objectY = interpolacaoQuadratica(startY, controlY, endY, t);  
  
    repaint();  
}
```

Run | Debug

```
public static void main(String[] args) {  
    SwingUtilities.invokeLater(new Runnable() {  
        public void run() {  
            AnimacaoInterpolacaoQuadratica animation = new AnimacaoInterpolacaoQuadratica();  
            animation.setVisible(b:true);  
        }  
    });  
}
```

A portrait of Isaac Newton with long, wavy hair, looking slightly to the right. The background is dark with various mathematical and scientific elements overlaid in a light green color. On the left, there is a diagram of a triangle with vertices labeled A and B, and a line segment labeled C. Above it, the text 'T. Trig. 3' is visible. In the center, the word 'OBRIGADO' is written in a large, serif font. Below it, the text ':)' is visible. On the right, there is a diagram of a circle with a point labeled F, and a line segment labeled P. Above it, the text 'E = ma' is visible. The overall image has a vintage, academic feel.

OBRIGADO

:)