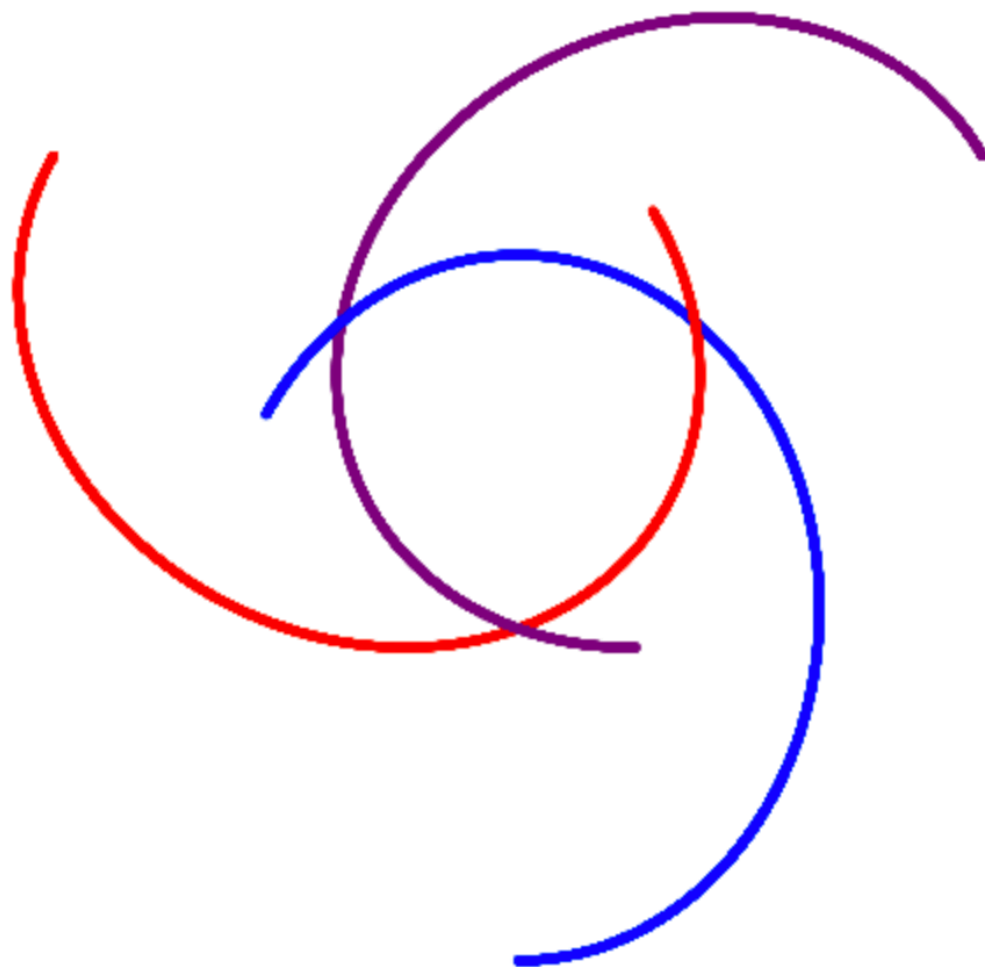


HTML5 课件——三体模拟

张皓通

功能演示



设计思路

界面布局

<head>

<body>

- 段落
- 超链接
- 列表
- button
- range
- Canvas

三体模拟器

[访问实验中心主页](#)

- 点击开始或停止按钮，控制动画
- 点击隐藏轨迹，质点的轨迹就会隐藏啦
- 点击太阳系、等边三角形，就会出现相应的稳定情况
- 移动滑块可以微调数值



设计思路

$$F_{12} = G \frac{m_1 m_2}{(r_1 - r_2)^2}$$

$$F_{23} = G \frac{m_2 m_3}{(r_2 - r_3)^2}$$

$$F_{31} = G \frac{m_3 m_1}{(r_3 - r_1)^2}$$

$$a_1 = \frac{F_{12} + F_{31}}{m_1}$$

$$a_2 = \frac{F_{12} + F_{23}}{m_2}$$

$$a_3 = \frac{F_{23} + F_{31}}{m_3}$$

$$v'_1 = v_1 + a_1 \times t$$

$$v'_2 = v_2 + a_2 \times t$$

$$v'_3 = v_3 + a_3 \times t$$

$$r'_1 = r_1 + v_1 \times t + \frac{1}{2} \times a_1 \times t^2$$

$$r'_2 = r_2 + v_2 \times t + \frac{1}{2} \times a_2 \times t^2$$

$$r'_3 = r_3 + v_3 \times t + \frac{1}{2} \times a_3 \times t^2$$

设计思路

JAVASCRIPT实现

- 定义全局变量

```
var animateButton = document.getElementById("animateButton"),
    paused = false;
var animateButton1 = document.getElementById("animateButton1"),
    trace = true;
var animateButton2 = document.getElementById("animateButton2"),
    special = 0;
var animateButton3 = document.getElementById("animateButton3");
var radiusRange = document.getElementById("radiusRange");

var inputm1 = document.getElementById("inputm1");
var inputm2 = document.getElementById("inputm2");
var inputm3 = document.getElementById("inputm3");

var myCanvas = document.getElementById("space");
var ball1 = myCanvas.getContext("2d");
var ball2 = myCanvas.getContext("2d");
var ball3 = myCanvas.getContext("2d");
var m = [0, 5, 5, 5];

var x1 = 100, y1 = 370;
var x2 = 200, y2 = 200;
var x3 = 300, y3 = 350;

var vx1 = 100, vy1 = 50;
var vx2 = 30, vy2 = -40;
var vx3 = 20, vy3 = -30;

var ax1 = 0, ay1 = 0;
var ax2 = 0, ay2 = 0;
var ax3 = 0, ay3 = 0;

var G = 6.67 * Math.pow(10, 4.5),
    t = 0.001;
```


设计思路

JAVASCRIPT实现

- Function gravity()
 - 在画布上画点
 - 运算得出t时间后小球位置
 - 再次调用gravity()，循环往复

```
ball1.beginPath();
ball1.arc(x1, y1, 2, 0, Math.PI * 2);
ball1.closePath();
ball1.fillStyle = "red";
ball1.fill();

ball2.beginPath();
ball2.arc(x2, y2, 2, 0, Math.PI * 2);
ball2.closePath();
ball2.fillStyle = "blue";
ball2.fill();

ball3.beginPath();
ball3.arc(x3, y3, 2, 0, Math.PI * 2);
ball3.closePath();
ball3.fillStyle = "purple";
ball3.fill();
```


设计思路

其他功能

- 开始/停止 —— 循环仍在进行，但是不会画点和更新位置
- 显示/隐藏轨迹 —— `ball.clearRect(0, 0, 1000, 600);`
- 特殊情况 —— 点击button，清空画布，并更新位置、速度信息
- 滑块

THANKS