

# QUARTO TESTE

Universidade Federal de Goiás (UFG) - Regional Jataí  
Bacharelado em Ciência da Computação  
Física para Ciência da Computação  
Esdras Lins Bispo Jr.

06 de novembro de 2019

## ORIENTAÇÕES PARA A RESOLUÇÃO

- A avaliação é individual, sem consulta;
- A pontuação máxima desta avaliação é 10,0 (dez) pontos, sendo uma das 06 (seis) componentes que formarão a média final da disciplina: quatro mini-testes (MT), uma prova final (PF) e eventuais exercícios-bônus propostos (EB);
- A média final ( $MF$ ) será calculada assim como se segue

$$MF = MIN(10, S)$$
$$S = \left[ \left( \sum_{i=1}^4 max(MT_i, SMT_i) + PF \right) \cdot 0,2 + EB \right]$$

em que

- $S$  é o somatório da pontuação de todas as avaliações, e
- $SMT_i$  é a substitutiva do mini-teste  $i$ .
- O conteúdo exigido compreende os seguintes pontos apresentados no Plano de Ensino da disciplina: (2) Movimentos, (3) Trabalho e Energia, (4) Colisões, (7) Projeto de Animação, e (8) Outros Tópicos.

Nome:
Assinatura:

1. (5,0 pt) **(Halliday 4.21 Adaptada)** Um dardo é arremessado horizontalmente com uma velocidade inicial de 20 m/s em direção a um ponto P, o centro de um alvo de parede. O dardo atinge um ponto Q do alvo, verticalmente abaixo de P, 0,4 s depois do arremesso. (Admita  $g = 10 \text{ m/s}^2$ )

- (a) Qual é a distância PQ?

**Resposta:** Basta calcular a distância percorrida da componente  $y$  em 0,4 s.

Distância PQ:.

$$\Delta y = v_0 t + \frac{1}{2} a t^2$$

$$\Delta y = 0 \text{ m/s} \times 0,4 \text{ s} + \frac{1}{2} (-10 \text{ m/s}^2) (0,4 \text{ s})^2$$

$$\Delta y = -0,8 \text{ m}$$

Logo, a distância PQ é 0,8 m. O sinal negativo indica o sentido do deslocamento.

- (b) A que distância do alvo foi arremessado o dardo?

**Resposta:** Basta calcular a distância percorrida da componente  $x$  em 0,4 s.

Distância do alvo:.

$$\Delta x = v t$$

$$\Delta x = 20 \text{ m/s} \times 0,4 \text{ s}$$

$$\Delta x = 8 \text{ m}$$

Logo a distância percorrida foi de 8 metros.

2. Em JavaScript, você irá escrever a função `emCadaPasso` de acordo com o modelo apresentado abaixo. Você deve substituir apenas as linhas 8, 11 e 13, pelos trechos de código 1, 2 e 3, respectivamente. O objetivo é que a bola azul realize um movimento oblíquo de forma que haja uma coeficiente de restituição de 80% ao colidir com as paredes laterais ou com o solo, simulando o comportamento de uma bola colidindo nas paredes de uma sala. O movimento da bola permanecerá indefinidamente, respeitando as condições já mencionadas.

```
1 function emCadaPasso() {
2     bola.x += bola.vx;
3     bola.vy += bola.ay;
4     bola.y += bola.vy;
5
6     if (bola.y > canvas.height - bola.raio){
7         bola.y = canvas.height - bola.raio;
8         // TRECHO 1
9     }
10    if(bola.x < bola.raio){
11        // TRECHO 2
12    }
13    if(/* TRECHO 3 */){
14        bola.x = canvas.width - bola.raio;
15        bola.vx = -bola.vx*0.8;
16    }
17
18    bola.desenhar(contexto);
19 }
```

### RESPOSTA:

```
1 //TRECHO 1
2 bola.vy = -bola.vy*0.8;
3
4 //TRECHO 2
5 bola.x = bola.raio;
6 bola.vx = -bola.vx*0.8;
7
8 //TRECHO 3
9 bola.x > canvas.width - bola.raio
```

# 1 Fórmulas Auxiliares

## 1.1 Movimento Uniforme (MU)

1.  $x = x_0 + vt$

## 1.2 Movimento Uniformemente Variado (MUV)

1.  $x - x_0 = v_0t + \frac{1}{2}at^2$

2.  $v = v_0 + at$

3.  $x - x_0 = \frac{1}{2}(v_0 + v)t$

4.  $x - x_0 = vt - \frac{1}{2}at^2$

5.  $v^2 = v_0^2 + 2a(x - x_0)$

## js/principal.js

```
1 var canvas = document.getElementById('canvas');
2 var contexto = canvas.getContext('2d');
3
4 var bola = new Bola(50, '#0000ff');
5 inicializar("emCadaPasso");
6
7 window.onload = init;
8
9 function init() {
10   setInterval(emCadaPasso, 1000/60);
11 };
```

## js/Bola.js

```
1 function Bola(raio, cor) {
2     this.raio = raio;
3     this.cor = cor;
4     this.x = 0;
5     this.y = 0;
6     this.vx = 0;
7     this.vy = 0;
8 }
9
10 Bola.prototype.desenhar = function (contexto) {
11     contexto.clearRect(0, 0, canvas.width, canvas.height);
12     contexto.fillStyle = this.cor;
13     contexto.beginPath();
14     contexto.arc(this.x, this.y, this.raio, 0, 2 * Math.PI,
15         true);
16     contexto.closePath();
17     contexto.fill();
18 };
```

## js/inicializacao.js

```
1 function bolabaseEsquerda(margem) {
2     bola.x = bola.raio + margem;
3     bola.y = canvas.height - bola.raio - margem;
4 }
5
6 function inicializar(valor){
7     switch(valor){
8         case "emCadaPasso":
9             bolaBaseEsquerda(30);
10             bola.vy = -120/60;
11             bola.vx = 60/60;
12             bola.ay = 98/60;
13             break;
14     }
15 }
```