

QUARTO TESTE

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

07 de março de 2017

ORIENTAÇÕES PARA A RESOLUÇÃO

- A avaliação é individual, sem consulta;
- A pontuação máxima desta avaliação é 10,0 (dez) pontos, sendo uma das 05 (cinco) componentes que formarão a média final da disciplina: dois testes, duas provas e exercícios-bônus;
- A média final (MF) será calculada assim como se segue

$$MF = MIN(10, S)$$
$$S = \left(\sum_{i=1}^4 0,2.T_i \right) + 0,2.P + EB$$

em que

- S é o somatório da pontuação de todas as avaliações,
 - T_i é a pontuação obtida no teste i ,
 - P é a pontuação obtida na prova, e
 - EB é a pontuação total dos exercícios-bônus.
- O conteúdo exigido compreende os seguintes pontos apresentados no Plano de Ensino da disciplina: (2) Medidas Físicas e Vetores, (3) Movimentos, (4) Trabalho e Energia, (5) Colisões, e (7) Outros Tópicos.

Nome:

Assinatura:

1. (5,0 pt) (**Halliday 4.21**) Um dardo é arremessado horizontalmente com uma velocidade inicial de 10 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, 0,19 s depois do arremesso.
 - (a) Qual é a distância PQ?
 - (b) A que distância do alvo foi arremessado o dardo?
2. Em JavaScript, adapte as funções `valoresIniciais` e `emCadaPasso` conforme apresentada abaixo. Você deve substituir apenas as linhas 7, 10 e 14, pelos trechos de código 1, 2 e 3, respectivamente. O objetivo é que uma bola faça o movimento de queda livre de cabeça para baixo (ao invés da bola cair para baixo, ela deve “cair para cima”). Lembre-se de que a bola quicará no teto até o repouso. Admita que em cada colisão com o teto, a bola transfere 20% de sua energia cinética. Admita também que velocidade inicial da bola, no eixo y, seja zero e que $g = 9,8 \text{ px/s}^2$.

```
1 function valoresIniciais() {  
2   this.bola = new Bola(50, '#0000ff');  
3  
4   this.bola.x = 100;  
5   this.bola.y = this.canvas.height - 50;  
6  
7   //TRECHO 1  
8 }  
9 function emCadaPasso() {  
10  if ( /* TRECHO 2 */ ) {  
11    this.bola.y += this.bola.vy;  
12    this.bola.vy += this.bola.ay;  
13  }  
14  // TRECHO 3  
15  
16  this.limparCanvas();  
17  this.bola.desenhar(this.contexto);  
18  this.bloco.desenhar(this.contexto);  
19 }
```

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

Protótipo Bola

```
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   this.ax = 0;
9   this.ay = 0;
10 }
11 function desenhara(contexto){
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 }
19 function dentroLimiteInferior(canvas) {
20   if (this.y + this.vy + this.raio <= canvas.height) {
21     return true;
22   } else {
23     return false;
24   }
25 }
```

```

23     }
24 }
25 function dentroLimiteSuperior(canvas) {
26     if (this.y + this.vy + this.raio <= 0) {
27         return true;
28     } else {
29         return false;
30     }
31 }
32 function dentroLimiteDireito(canvas) {
33     if (this.x + this.vx + this.raio <= canvas.width) {
34         return true;
35     } else {
36         return false;
37     }
38 }
39 Bola.prototype.desenhar = desenhar;
40 Bola.prototype.dentroLimiteInferior = dentroLimiteInferior;
41 Bola.prototype.dentroLimiteSuperior = dentroLimiteSuperior;
42 Bola.prototype.dentroLimiteDireito = dentroLimiteDireito;

```

Protótipo Bloco

```

1 function Bloco(largura, altura, cor) {
2     this.largura = largura;
3     this.altura = altura;
4     this.cor = cor;
5     this.x = 0;
6     this.y = 120;
7 }
8 function desenhar(contexto){
9     contexto.fillStyle = this.cor;
10    contexto.beginPath();
11    contexto.rect(this.x, this.y, this.largura, this.altura);
12    contexto.closePath();
13    contexto.fill();
14 }
15 Bloco.prototype.desenhar = desenhar;

```

Protótipo Ambiente

```
1 function Ambiente() {}
2 function iniciar() {
3     this.canvas = document.getElementById('canvas');
4     this.contexto = this.canvas.getContext('2d');
5
6     this.valoresIniciais();
7
8     setInterval(this.emCadaPasso, 1000 / 60); // 60 fps
9 }
10 function valoresIniciais() {
11     this.bola = new Bola(50, '#0000ff');
12
13     this.bola.x = 210;
14     this.bola.y = 70;
15
16     this.bola.vx = 55 / 60;
17     this.bola.vy = 0;
18
19     this.bola.ax = 0;
20     this.bola.ay = 9.8 / 60;
21 }
22 function emCadaPasso() {
23     if (this.bola.dentroLimiteInferior(this.canvas)) {
24         this.bola.y += this.bola.vy;
25         this.bola.vy += this.bola.ay;
26     }
27
28     this.limparCanvas();
29     this.bola.desenhar(this.contexto); // desenhe a bola
30     this.bloco.desenhar(this.contexto);
31 }
32 function limparCanvas() {
33     this.contexto.clearRect(0, 0, this.canvas.width, this.
34         canvas.height);
35 }
36 Ambiente.prototype.iniciar = iniciar;
37 Ambiente.prototype.valoresIniciais = valoresIniciais;
38 Ambiente.prototype.emCadaPasso = emCadaPasso;
39 Ambiente.prototype.limparCanvas = limparCanvas;
```

Arquivo Principal

```
1 var amb = new Ambiente();  
2 window.onload = amb.iniciar;
```