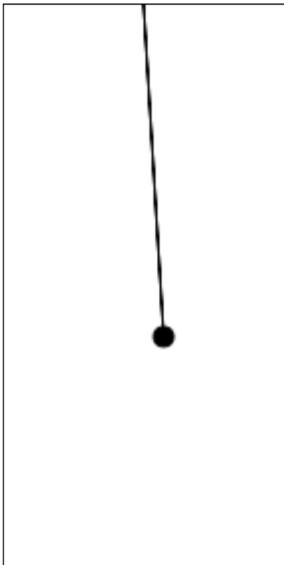


27 Pêndulo simples

O quadro a seguir mostra o que poderíamos chamar de um "experimento virtual", no sentido em que é possível fazer medidas realistas com ele. O movimento do pêndulo é controlado por um temporizador que utiliza o tempo real para posicioná-lo, assim como o cronômetro, que de fato apresenta o tempo decorrido entre o "liga" e o "desliga".

Desse modo, pode-se iniciar o cronômetro e começar simultaneamente a contar as oscilações, e desligar o cronômetro após, digamos, 10 oscilações. Dividindo o tempo decorrido por 10, no caso, obtém-se o período do pêndulo, a partir do qual é possível calcular o seu comprimento com a fórmula dada.

O quadro ainda disponibiliza uma calculadora para auxiliar o usuário a fazer as contas e escolher a alternativa que mais se aproxima do valor calculado.



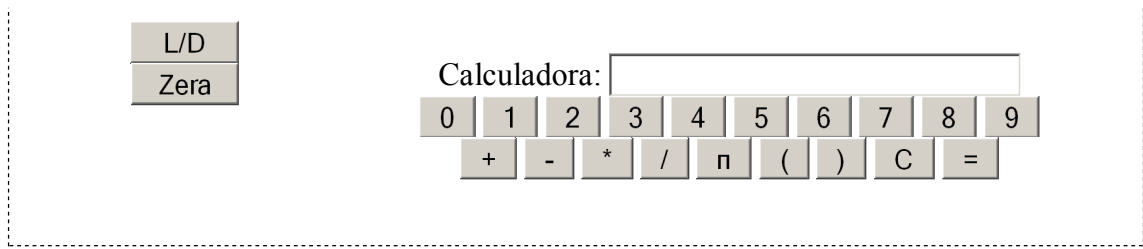
O período de um pêndulo simples é dado por $T = 2\pi(L/g)^{1/2}$, onde L é o seu comprimento e $g = 9,8 \text{ m/s}^2$ é a aceleração da gravidade. Podemos isolar o comprimento e obter:

$$L = g(T/2\pi)^2$$

Utilize o cronômetro abaixo e a fórmula acima para determinar o comprimento do pêndulo ao lado.

Qual a sua resposta?

- 1.9 m
- 2.1 m
- 2.3 m
- 1.5 m
- 1.7 m



O quadro foi construído como uma tabela com uma linha e duas colunas. Na primeira coluna, estão um canvas para o pêndulo e os elementos `input` do tipo `text` e `button` para implementar o cronômetro. Na segunda coluna, estão o texto da questão, um conjunto de `input` do tipo `radio` com as respostas e um `input` do tipo `text` para a expressão a ser processada pela calculadora e `input` do tipo `button`, que constróem os botões da calculadora.

A calculadora, em particular, foi implementada como segue:

```
<p style="text-align:center">Calculadora:
<input type="text" size="30" id="calc" value=""><br>
<input type="button" value="0" onClick="calcBut('0')">
<input type="button" value="1" onClick="calcBut('1')">
<input type="button" value="2" onClick="calcBut('2')">
<input type="button" value="3" onClick="calcBut('3')">
<input type="button" value="4" onClick="calcBut('4')">
<input type="button" value="5" onClick="calcBut('5')">
<input type="button" value="6" onClick="calcBut('6')">
<input type="button" value="7" onClick="calcBut('7')">
<input type="button" value="8" onClick="calcBut('8')">
<input type="button" value="9" onClick="calcBut('9')"><br>
<input type="button" value="+" onClick="calcBut('+')">
<input type="button" value="-" onClick="calcBut('-')">
<input type="button" value="*" onClick="calcBut('*')">
<input type="button" value="/" onClick="calcBut('/')">
<input type="button" value="n" onClick="calcBut('p')">
<input type="button" value="(" onClick="calcBut('(')(">
<input type="button" value=")" onClick="calcBut(')')(">
<input type="button" value="C" onClick="calcBut('C')">
<input type="button" value="=" onClick="calcBut('=')">
</p>
```

A função `calcBut()`, que é chamada quando o usuário pressiona qualquer um dos botões, foi definida como:

```
function calcBut(code) {
  var str = document.getElementById('calc').value;
  switch (code) {
    case '0': str += '0'; break;
    case '1': str += '1'; break;
    case '2': str += '2'; break;
    case '3': str += '3'; break;
    case '4': str += '4'; break;
    case '5': str += '5'; break;
    case '6': str += '6'; break;
    case '7': str += '7'; break;
    case '8': str += '8'; break;
    case '9': str += '9'; break;
```

```

        case '+': str += '+'; break;
        case '-': str += '-'; break;
        case '*': str += '*'; break;
        case '/': str += '/'; break;
        case '(': str += '('; break;
        case ')': str += ')'; break;
        case 'p': str += 'Math.PI'; break;
        case 'C': str = ""; break;
        case '=': str = eval(str); break;
    }
    document.getElementById('calc').value = str;
}

```

Os códigos para implementar um canvas e um cronômetro elementar já foram vistos em outros capítulos, assim como a construção de questões utilizando conjuntos de botões tipo radio.

Exercícios

1. Neste exemplo, o desafio está em fazer um algoritmo para a construção da resposta à questão. Ele deve ter algumas características:
 - a. Os valores gerados são próximos ao valor correto, mas não o bastante para que erros de arredondamento dificultem a escolha.
 - b. A resposta correta é colocada em uma ordem aleatoriamente escolhida cada vez que o documento é carregado, de modo que não ocorram respostas "viciadas".