

Criei um algoritmo que faça os cálculos par diversas quantidades de divisões diferentes. Como pode ser visto com 100 divisões já obtemos erros extremamente baixos, ainda assim é possível reduzir esse valor ao aumentar o número de divisões. Fiz apenas até com 700 partes para que o s resultados coubessem em um único print.

```
1 K=9e9
2
3 q=1e-6
4
5 for partes = 100:100:700
6     fprintf('Quantidade de partes: %d', partes);
7     disp(' ');
8
9     x=[2:(1/partes):3];
10
11     deltaQ=q/partes;
12
13     deltaE=K.*deltaQ./(x.^2);
14
15     SolNum=sum(deltaE)
16
17     exatoE=K*q/(2*(2+1))
18
19     ErroPercentual=(SolNum-exatoE)/exatoE*100;
20     fprintf('Erro percentual = %d%%', ErroPercentual);
21     disp(' ');
22     disp(' ');
23 end
```

```
octave:1> ex1
K = 9.0000e+09
q = 1.0000e-06
Quantidade de partes: 100
SolNum = 1516.3
exatoE = 1500
Erro percentual = 1.08421%

Quantidade de partes: 200
SolNum = 1508.1
exatoE = 1500
Erro percentual = 0.541887%

Quantidade de partes: 300
SolNum = 1505.4
exatoE = 1500
Erro percentual = 0.361209%

Quantidade de partes: 400
SolNum = 1504.1
exatoE = 1500
Erro percentual = 0.270888%

Quantidade de partes: 500
SolNum = 1503.3
exatoE = 1500
Erro percentual = 0.216702%

Quantidade de partes: 600
SolNum = 1502.7
exatoE = 1500
Erro percentual = 0.18058%

Quantidade de partes: 700
SolNum = 1502.3
exatoE = 1500
Erro percentual = 0.15478%
```