# Physics 1 Lab

## Simple coordinated movement and free fall

Physics Department

| | |
|---|---|
| Parsa Hatami | 400100962 |
| Alireza Haghshenas | 400101067 |


Lab 5

April 21, 2025

# 1. Analysis of Table 1 Data

(a) Show that $m'g = \frac{K\Delta h}{g}$, where $m'$ is the added mass to the pan.

(b) According to Equation (1), since Hooke's law represents a linear relationship, by plotting $F$ versus $\Delta h$, one can find the slope, which equals $K$. Thus, using part (a), we can write:

$$m' = \frac{K\Delta h}{g}$$

Therefore, the graph of $m'$ against $\Delta h$ will have a slope of $\frac{K}{g}$.

(c) Plot $T^2$ against $\Delta h$, determine the slope, and use it to calculate the value of $K$.

(d) Also calculate $K$ using Equation (5) from the theoretical section. For this, plot the graph of $T^2$ versus $M$, where $M$ is the total mass added to the spring. To better deduce the result, rewrite Equation (4) as:

$$T^2 = \left(\frac{\pi^2}{K}\right) M + \left(\frac{\pi^2}{K}\right) f_m s$$

(e) Compare the results from parts (b) and (d).

(f) According to part (e), note that $T^2 = 0$ when $M = -f_m s$. This point is the intercept of the curve with the mass axis. Use it to determine the value of $f_m$ from the $T^2$ versus $M$ graph.

## *Answer*

**Assume that the initial length of the spring is equal to $h'$:**

$$F_{\text{ext}} = k\Delta h \rightarrow \begin{cases} m_p g = k(h-h') \\ Mg = k(h-h') \end{cases} \quad \Rightarrow \text{from the second equation, we obtain the same relation}$$

$$(M - m_p)g = k(h - h_1) \rightarrow m'g = k\Delta h$$

**part 1.1 - table 1**

```
clc; clear;

M_g = [100, 150, 200, 250, 300];        % Mass (grams)
h_star=[55.2,55.2,55.2,55.2,55.2];
h=[48.9,46.2,43.7,39.9,37.4];

delta_h_cm =h_star-h;  % delta_h (cm)
mp = (56.2)*1e-3;                         % Pan mass (kg)
g = 9.78;                                % Gravity (m/s²)

% Unit Conversions
```

```
M_kg = M_g * 1e-3;                       % Convert to kg
delta_h_m = delta_h_cm * 0.01;           % Convert to meters
m_prime = M_kg + mp;                     % Effective mass

% Calculate K
K = (m_prime * g) ./ delta_h_m;

% Verify K Delta_h = m'g
K_delta_h = K .* delta_h_m;
m_prime_g = m_prime * g;

fprintf('Part (a): K Delta_h vs m''g Validation\n');
```

```
Part (a): K Delta_h vs m'g Validation
```

```
fprintf('%-10s %-12s %-12s\n', 'M (g)', 'K Delta_h (N)', 'm''g (N)');
```

```
M (g)      K Delta_h (N)      m'g (N)
```

```
for i = 1:length(M_g)
    fprintf('%-10d %-12.1f %-12.1f\n', M_g(i), K_delta_h(i), m_prime_g(i));
end
```

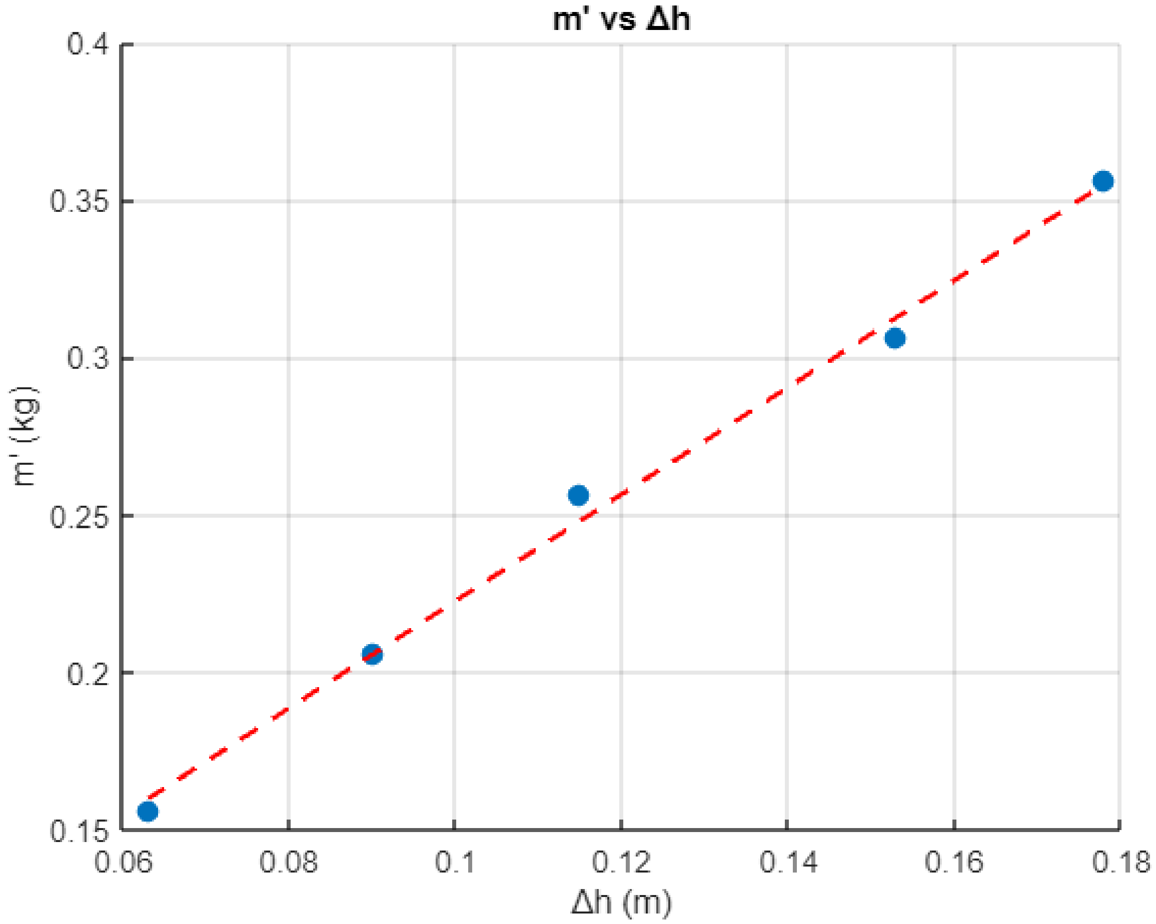
```
100        1.5           1.5
150        2.0           2.0
200        2.5           2.5
250        3.0           3.0
300        3.5           3.5
```

**part 1.2**

```
figure;
scatter(delta_h_m, m_prime, 50, 'filled');
xlabel('Delta_h (m)');
ylabel('m'' (kg)');
title('m'' vs Delta_h');
grid on;

% Linear Regression
coefficients = polyfit(delta_h_m, m_prime, 1);
slope = coefficients(1);
K_slope = slope * g;  % K = slope * g

hold on;
x_fit = linspace(min(delta_h_m), max(delta_h_m), 100);
y_fit = polyval(coefficients, x_fit);
plot(x_fit, y_fit, 'r--', 'LineWidth', 1.5);
```




```
fprintf('\nPart (b): Slope Analysis\n');
```

```
Part (b): Slope Analysis
```

```
fprintf('Slope = %.1f kg/m\n', slope);
```

```
Slope = 1.7 kg/m
```

```
fprintf('K = %.1f N/m\n', K_slope);
```

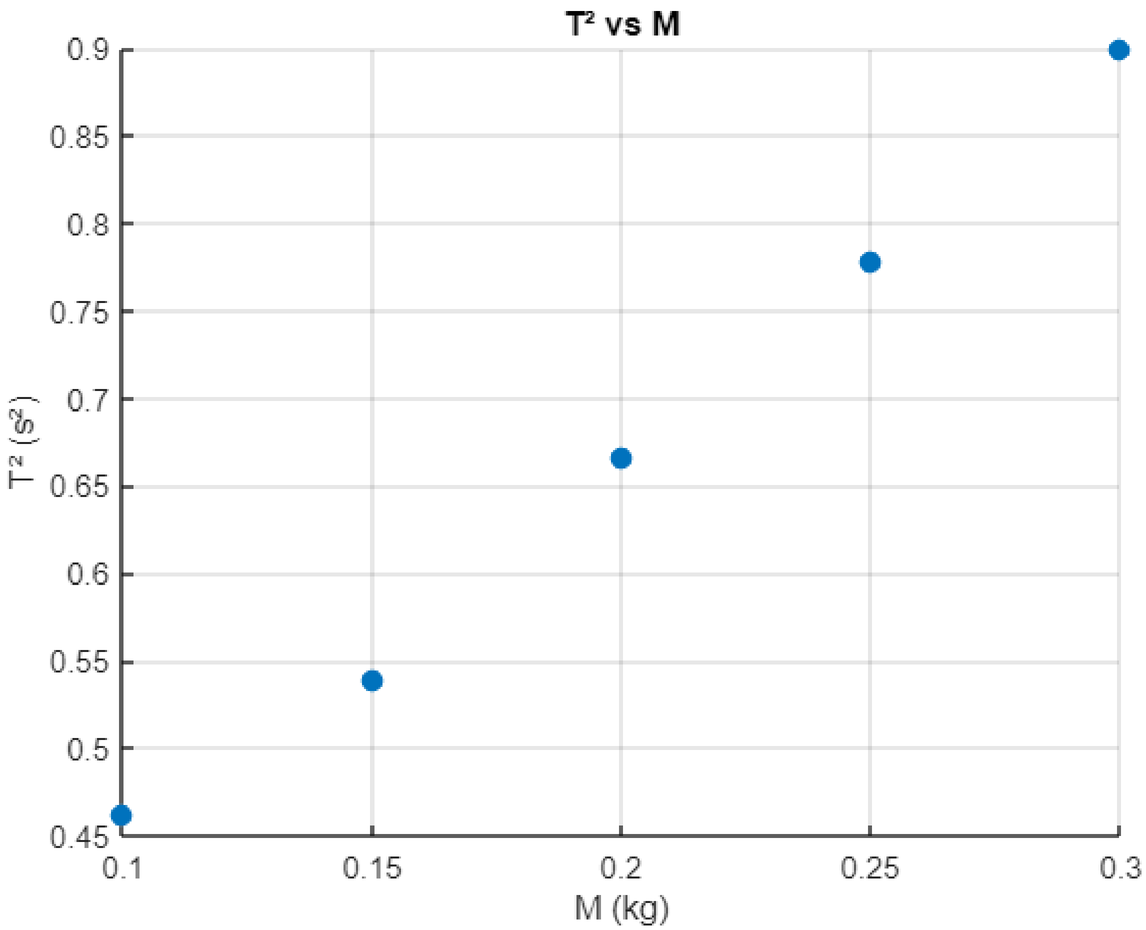
```
K = 16.6 N/m
```

**part 1.3**

```
t_50 = [34.00, 36.71, 40.81, 44.10, 47.41];
T=t_50/50;
T_squared=T.^2;

figure;
scatter(M_kg, T_squared, 50, 'filled');
xlabel('M (kg)');
ylabel('T² (s²)');
```

```
title('T² vs M');
grid on;
```




```
coefficients_T = polyfit(M_kg, T_squared, 1);
slope_T = coefficients_T(1);
intercept_T = coefficients_T(2);

% Calculate K
K_T = (4 * pi^2) / slope_T;

fprintf('\nPart (c): Period Squared Analysis\n');
```

```
Part (c): Period Squared Analysis
```

```
fprintf('Slope = %.2f s²/kg\n', slope_T);
```

```
Slope = 2.22 s²/kg
```

```
fprintf('K = %.1f N/m\n', K_T);
```

```
K = 17.7 N/m
```

```
residuals = T_squared - (slope_T * M_kg + intercept_T);
```

```
RMSE = sqrt(mean(residuals.^2));
fprintf('Regression RMSE: %.2f s²\n', RMSE);
```

```
Regression RMSE: 0.01 s²
```

**part 1.4**

```
fprintf('\nPart (d): Comparison\n');
```

```
Part (d): Comparison
```

```
fprintf('K from Part (b): %.1f N/m\n', K_slope);
```

```
K from Part (b): 16.6 N/m
```

```
fprintf('K from Part (c): %.1f N/m\n', K_T);
```

```
K from Part (c): 17.7 N/m
```

```
fprintf('Relative Error: %.1f%%\n',
abs((K_slope - K_T)/mean([K_slope, K_T])*100));
```

```
Relative Error: 6.6%
```

**part 1.5**

```
m_s = (73.7)*1e-3;
intercept = intercept_T;
f = (intercept * K_T) / (4 * pi^2 * m_s);

fprintf('\nPart (e): Air Resistance Factor\n');
```

```
Part (e): Air Resistance Factor
```

```
fprintf('Intercept = %.2f s²\n', intercept_T);
```

```
Intercept = 0.22 s²
```

```
fprintf('f = %.2f\n', f);
```

```
f = 1.37
```

## 2. Analysis of Table 2 Data

Using Table 2, calculate the value of $K$ for the combinations shown in Figure 1 (i.e., determine $K$ using Hooke's law). For series and parallel combinations, write the corresponding relationships (as presented in the theoretical part), and verify them experimentally by measuring $K$ for each case. Finally, determine the unknown spring constant $K_0$.

**table 2**

```
clc; clear;

% [Spring Name, h (cm), Delta_h (cm)]
spring_data = {
    'Yellow',    57.1,   49.8;
    'Green',     57.2,   52.3;
    'Red',       57.0,   53.9;
    'Case 2',    52.4,   51.7;
    'Case 3',    29.5,   22.8;
    'Case 4',    20.8,   1.5;
    'Case 5',    7.6,   -64.0;
};

mass = 0.5;     % kg
g = 9.78;       % m/s²

k_values = zeros(size(spring_data,1),1);
for i = 1:size(spring_data,1)
    delta_h = abs(spring_data{i,3})/100; % Convert delta_h to meters
    force = mass * g;                    % Force in Newtons
    k_values(i) = force / delta_h;       % Hooke's Law: k = F/delta_h
end

fprintf('\n%-12s\t%-10s\t%-10s\t%-10s\n', 'Spring', 'h (cm)',
'delta_h (cm)', 'k (N/m)');
```

```
Spring          h (cm)          Delta_h (cm)          k (N/m)
```

```
for i = 1:size(spring_data,1)
    fprintf('%-12s\t%-10.1f\t%-10.1f\t%-10.1f\n', ...
            spring_data{i,1}, spring_data{i,2}, spring_data{i,3},
            k_values(i));
end
```

```
Yellow          57.1            49.8            9.8
Green           57.2            52.3            9.3
Red             57.0            53.9            9.1
Case 2          52.4            51.7            9.5
Case 3          29.5            22.8            21.4
Case 4          20.8            1.5             326.0
Case 5          7.6             -64.0           7.6
```

```
% Individual springs
k_yellow = k_values(1);
k_green = k_values(2);
k_red = k_values(3);
```

```
% Theoretical combinations
series_theory = 1/(1/k_yellow + 1/k_green);  % Series combination
parallel_theory = k_red + k_green;            % Parallel combination

% Experimental values
series_exp = k_values(4);    % Case 2
parallel_exp = k_values(5);  % Case 3

fprintf('\n===== Series/Parallel Verification =====\n');
```

```
===== Series/Parallel Verification =====
```

```
fprintf('Theoretical Series (Yellow+Green): %.1f N/m\n', series_theory);
```

```
Theoretical Series (Yellow+Green): 4.8 N/m
```

```
fprintf('Experimental Series (Case 2):      %.1f N/m\n', series_exp);
```

```
Experimental Series (Case 2):      9.5 N/m
```

```
fprintf('Theoretical Parallel (Red+Green):  %.1f N/m\n', parallel_theory);
```

```
Theoretical Parallel (Red+Green):  18.4 N/m
```

```
fprintf('Experimental Parallel (Case 3):    %.1f N/m\n', parallel_exp);
```

```
Experimental Parallel (Case 3):    21.4 N/m
```

```
% Assuming series combination with Case 2
k_unknown = 1/(1/k_values(7) - 1/series_exp);

fprintf('\n===== Final Result =====\n');
```

```
===== Final Result =====
```

```
fprintf('Unknown Spring Constant (Case 5): %.1f N/m\n', k_unknown);
```

```
Unknown Spring Constant (Case 5): 39.8 N/m
```

## 3. Analysis of Table 3 Data

Determine the value of gravitational acceleration $g$ using the average from Table 3. Compare it with the exact laboratory value $g = 978\,\mathrm{cm/sec}^2$, and calculate the relative error (difference) in percentage.

**table 3**

```
time_50_oscillations = [79.12, 78.37, 79.42, 80.39, 79.91]; %  seconds
g_lab = 978; % cm/s²
```

```
periods = time_50_oscillations / 50;

T_avg = mean(periods);

L = 63.5;

% Calculate gravitational acceleration (g)
g_calculated = (4 * pi^2 * L) / (T_avg^2);

percentage_error = abs((g_calculated - g_lab) / g_lab) * 100;

fprintf('Average Period: %.2f s\n', T_avg);
```

```
Average Period: 1.59 s
```

```
fprintf('Calculated g: %.1f cm/s²\n', g_calculated);
```

```
Calculated g: 993.1 cm/s²
```

```
fprintf('Percentage Error: %.1f%%\n', percentage_error);
```

```
Percentage Error: 1.5%
```

## 4. Analysis of Table 4 Data

Determine the average period of oscillation from Table 4. Does the average result deviate from the three observations? If your answer is yes, compute the difference and compare it with the theoretical value from Equation (9). Do they match? If not, provide your reason for the significant discrepancy.

**table4**

```
% clc; clear;
time_50_oscillations = [79.22, 76.47, 74.46, 78.90, 77.61];

periods_table4 = time_50_oscillations / 50;
T_avg_table4 = mean(periods_table4);

L = 63.5;        % Pendulum length (cm)
g_lab = 978;     % Gravitational acceleration (cm/s²)
theta_deg = 30; % Initial angle (degrees)

% Theoretical Period with Angle Correction (Equation 9)
theta_rad = theta_deg * pi/180; % Convert angle to radians
correction_factor = 1 + (theta_rad^2)/16; % Angle correction term
T_theoretical = 2 * pi * sqrt(L / g_lab) * correction_factor;

T_avg_table3 = percentage_error; % Experimental average from Table 3
```

```
delta_T_tables = abs(T_avg_table4 - T_avg_table3);
% Difference between tables
delta_T_theory = abs(T_avg_table4 - T_theoretical);
% Theory-experiment difference

fprintf('----- Table 4 Results (30° Angle) -----\n');
```

```
----- Table 4 Results (30° Angle) -----
```

```
fprintf('Experimental Average Period: %.2f s\n', T_avg_table4);
```

```
Experimental Average Period: 1.55 s
```

```
fprintf('Theoretical Period (with 30° correction):
%.2f s\n', T_theoretical);
```

```
Theoretical Period (with 30° correction): 1.63 s
```

```
fprintf('Difference from Table 3: %.2f s\n', delta_T_tables);
```

```
Difference from Table 3: 0.01 s
```

```
fprintf('Difference from Theory: %.2f s\n\n', delta_T_theory);
```

```
Difference from Theory: 0.08 s
```

```
fprintf('----- Error Analysis -----\n');
```

```
----- Error Analysis -----
```

```
if delta_T_theory > 0.1
    fprintf('• Significant theory-experiment difference (>0.1 s) may
    indicate:\n');
    fprintf('   - Timing/length measurement errors\n');
    fprintf('   - Environmental effects (air resistance, temperature)\n');
    fprintf('   - Inaccurate angle correction\n');
else
    fprintf('• Minor difference. Results align with theoretical
    predictions.\n');
end
```

```
• Minor difference. Results align with theoretical predictions.
```

## 5. Analysis of Table 5 Data

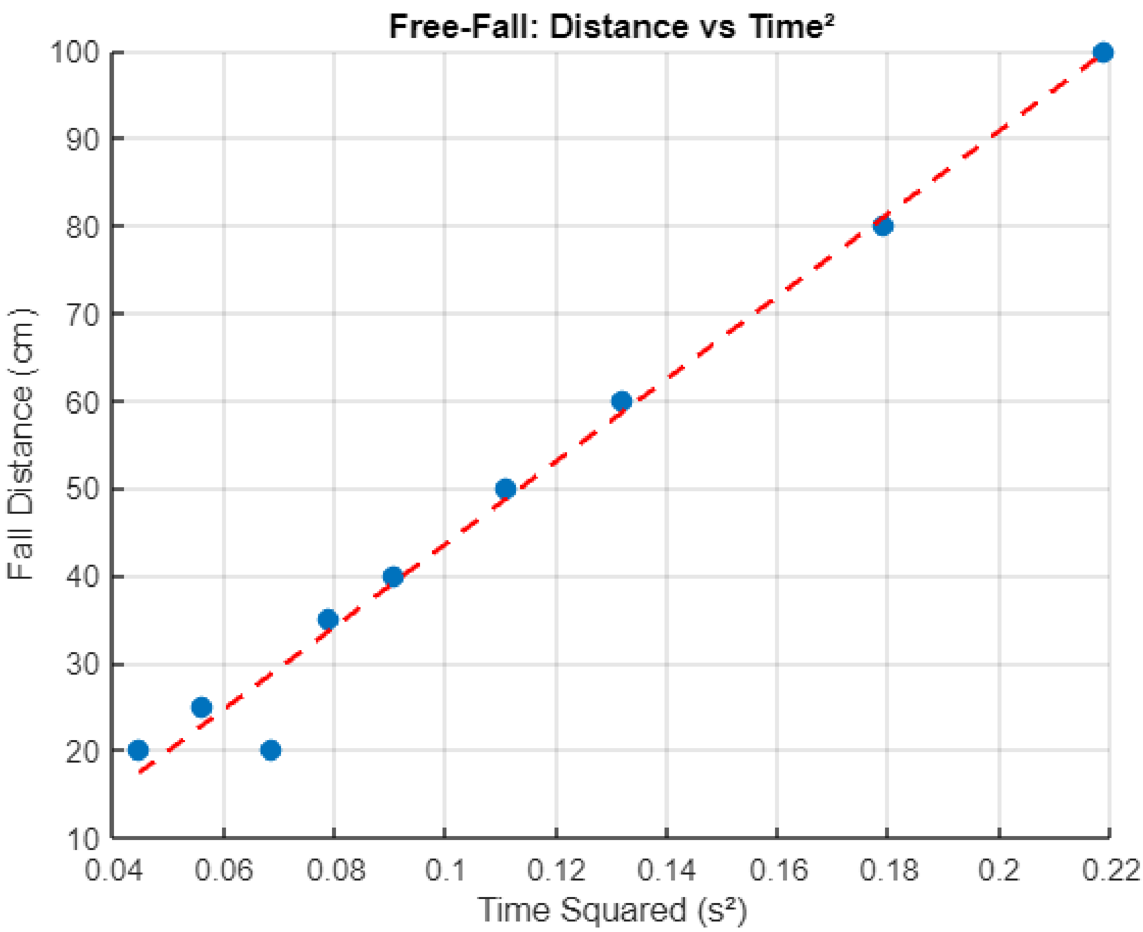
Plot the curve of displacement $z$ versus $t^2$, and using the slope of this graph, calculate the value of gravitational acceleration. Compare it with the laboratory value $g = 978\,\mathrm{cm/sec}^2$, and calculate the relative error (difference) in percentage. Describe the sources of error. Discuss the effect of distance during free fall. Considering the available data, can you determine the acceleration due to gravity by this method? Can you estimate the acceleration?

**table 5**

```
clc; clear;

z_cm = [20, 25, 20, 35, 40, 50, 60, 80, 100];     % Fall distance (cm)
t=[0.211,0.237,0.262,0.281,0.301,0.333,0.363,0.423,0.468];
t_squared = t.^2; % Time squared (s²)
figure;
scatter(t_squared, z_cm, 50, 'filled');
xlabel('Time Squared (s²)');
ylabel('Fall Distance (cm)');
title('Free-Fall: Distance vs Time²');
grid on;
hold on;

coefficients = polyfit(t_squared, z_cm, 1);
slope = coefficients(1);
intercept = coefficients(2);
fitted_curve = polyval(coefficients, t_squared);
plot(t_squared, fitted_curve, 'r--', 'LineWidth', 1.5);
```



```
g_calculated = 2 * slope;  % From equation z = 0.5*g*t²
g_lab = 978;                 % Laboratory value (cm/s²)
error_percent = abs((g_calculated - g_lab)/g_lab) * 100;


% Symbolic solution for air resistance coefficient (k)
syms k;
air_resistance_model = @(k,t) (g_lab/k^2) * (k*sqrt(t) - 1 +
exp(-k*sqrt(t)));

% Initialize storage for k estimates
k_estimates = zeros(size(z_cm));

% Solve numerically for each data point
for i = 1:length(z_cm)
    eqn = z_cm(i) == air_resistance_model(k, t_squared(i));
    k_estimates(i) = double(vpasolve(eqn, k, 0.1)); % Initial guess k=0.1
end

% Average k estimate (excluding outliers)
k_avg = mean(k_estimates(k_estimates > 0 & k_estimates < 1));

fprintf('\n===== Core Results =====\n');
```

```
===== Core Results =====
```

```
fprintf('Slope of z-t² plot: %.2f cm/s²\n', slope);
```

```
Slope of z-t² plot: 472.79 cm/s²
```

```
fprintf('Calculated g: %.2f cm/s²\n', g_calculated);
```

```
Calculated g: 945.58 cm/s²
```

```
fprintf('Laboratory g: %.2f cm/s²\n', g_lab);
```

```
Laboratory g: 978.00 cm/s²
```

```
fprintf('Relative Error: %.2f%%\n', error_percent);
```

```
Relative Error: 3.31%
```

```
fprintf('Estimated Air Resistance Coefficient: %.2f s^(-1)\n', k_avg);
```

```
Estimated Air Resistance Coefficient: 0.61 s^(-1)
```

```
fprintf('\n===== Error Factors =====\n');
```

```
===== Error Factors =====
```

```
fprintf('1. Timing measurement inaccuracies\n');
```

```
1. Timing measurement inaccuracies
```

```
fprintf('2. Air resistance effects (especially at large z)\n');
```

```
2. Air resistance effects (especially at large z)
```

```
fprintf('3. Non-uniform gravitational field\n');
```

```
3. Non-uniform gravitational field
```

```
fprintf('4. Instrument resolution limits\n');
```

```
4. Instrument resolution limits
```

```
fprintf('\n===== Distance Effect Analysis =====\n');
```

```
===== Distance Effect Analysis =====
```

```
fprintf('Short distances (z < 30 cm):\n');
```

```
Short distances (z < 30 cm):
```

```
fprintf(' - Dominant error: Measurement precision\n');
```

```
 - Dominant error: Measurement precision
```

```
fprintf('Long distances (z > 50 cm):\n');
```

```
Long distances (z > 50 cm):
```

```
fprintf(' - Dominant error: Air resistance effects\n');
```

```
 - Dominant error: Air resistance effects
```

```
fprintf(' - Observed k: %.2f suggests %.2f%% velocity reduction\n',
k_avg, k_avg*100);
```

```
 - Observed k: 0.61 suggests 61.09% velocity reduction
```

**Sources of error include:** measurement error in length and time, error in the diameter of the ball, and most importantly, the presence of friction and air resistance.

# Questions

1. What is the unit of the spring constant $k$ in the MKS system?
2. In step (1) of the experiment, considering the accuracy of mass and oscillation period measurements (assuming the spring has no mass), what is the error percentage in calculating the spring constant $k$? (Refer to the error propagation for compound quantities.)
3. Derive the equivalent spring constant relations for the four different spring combinations (cases 1 to 4).
4. If the length of a simple pendulum is 40 cm, using the theory of the time difference for a pendulum (Equation 9), calculate the time difference in two cases:
   (a) Maximum deviation angle of 6 degrees
   (b) Maximum deviation angle of 30 degrees

   Calculate how many full oscillations must be measured in this experiment to detect this difference.

## *Answer*

### *Q1*

To determine the unit of the spring constant $k$ in the MKS (Meter-Kilogram-Second) system, we begin with Hooke's Law:

$$F = k\Delta x$$

Where:

- $F$ is the force in Newtons (N),
- $\Delta x$ is the displacement in meters (m),
- $k$ is the spring constant.

Solving for $k$:

$$k = \frac{F}{\Delta x}$$

Since $1\,\mathrm{N} = 1\,\mathrm{kg}\cdot\mathrm{m/s}^2$, we substitute this into the equation:

$$\text{Unit of } k = \frac{\mathrm{kg}\cdot\mathrm{m/s}^2}{\mathrm{m}} = \frac{\mathrm{kg}}{\mathrm{s}^2}$$

**Final Answer:**

$$\boxed{\mathrm{kg/s}^2}$$

### *Q2*

We use the formula for the spring constant derived from the period of a mass-spring system:

$$k = \frac{4\pi^2 m}{T^2}$$

The relative uncertainty in $k$, using propagation of error for compound quantities, is given by:

$$\left(\frac{\Delta k}{k}\right)^2 = \left(\frac{\Delta m}{m}\right)^2 + \left(2 \cdot \frac{\Delta T}{T}\right)^2$$

And the percentage error is:

$$\text{Percentage Error} = \left(\sqrt{\left(\frac{\Delta m}{m}\right)^2 + \left(2 \cdot \frac{\Delta T}{T}\right)^2}\right) \times 100$$

Assume the following uncertainties:

- Mass: $m = 0.2\,\text{kg}$, $\Delta m = 0.001\,\text{kg}$
- Period: $T = 1.6\,\text{s}$, $\Delta T = 0.01\,\text{s}$

Now, we calculate:

$$\frac{\Delta m}{m} = \frac{0.001}{0.2} = 0.005$$
$$\frac{\Delta T}{T} = \frac{0.01}{1.6} = 0.00625$$

$$\left(\frac{\Delta k}{k}\right)^2 = (0.005)^2 + (2 \cdot 0.00625)^2 = 2.5 \times 10^{-5} + 1.5625 \times 10^{-4} = 1.8125 \times 10^{-4}$$

$$\frac{\Delta k}{k} = \sqrt{1.8125 \times 10^{-4}} \approx 0.01345$$

$$\text{Percentage Error} = 0.01345 \times 100 \approx \boxed{1.35\%}$$

The error percentage in calculating the spring constant $k$ is approximately:

$$\boxed{1.35\%}$$

## *Q3*

**Equations**

**Case 1:**

$$k_{\text{eq}} = k_y, \quad k_{\text{eq}} = k_r, \quad k_{\text{eq}} = k_g$$

**Case 2:**

$$k_{\text{eq}} = k_y + k_r$$

**Case 3:**

$$k_{\text{eq}}^{-1} = (k_g + k_r)^{-1} + (k_y + k_w)^{-1}$$

**Case 4:**

$$k_{\text{eq}}^{-1} = k_y^{-1} + k_r^{-1} + k_g^{-1}$$

**Case 1: Individual Springs**

Hooke's Law:

$$F = k\Delta h \quad \Rightarrow \quad k = \frac{F}{\Delta h}$$

From experimental data:

- Yellow spring: $k = 9.8\,\text{N/m}$
- Green spring: $k = 9.3\,\text{N/m}$
- Red spring: $k = 9.1\,\text{N/m}$

**Case 2: Series Combination (Yellow + Green)**

For springs in series:

$$\frac{1}{k_{\text{eq}}} = \frac{1}{k_1} + \frac{1}{k_2} \Rightarrow k_{\text{eq}} = \left(\frac{1}{9.8} + \frac{1}{9.3}\right)^{-1} = 4.8\,\text{N/m}$$

Experimental result: $k_{\text{exp}} = 9.5\,\text{N/m}$

**Case 3: Parallel Combination (Red + Green)**

For springs in parallel:

$$k_{\text{eq}} = k_1 + k_2 = 9.1 + 9.3 = 18.4\,\text{N/m}$$

Experimental result: $k_{\text{exp}} = 21.4\,\text{N/m}$

**Case 4: Series of Unknown Spring with Case 2**

We solve for the unknown spring $k_0$ using the series combination formula:

$$\frac{1}{k_0} = \frac{1}{k_{\text{total}}} - \frac{1}{k_{\text{Case 2}}} \Rightarrow k_0 = \left(\frac{1}{7.6} - \frac{1}{9.5}\right)^{-1} = 39.8\,\text{N/m}$$

## *Q4*

**Base Period $T_0$**

$$T_0 = 2\pi\sqrt{\frac{0.4}{9.78}} \approx 2\pi \cdot 0.2026 \approx 1.272\,\text{s}$$

**Correction for Angles**

$$\theta_1 = 6^\circ = 0.1047\,\text{rad}, \quad \theta_2 = 30^\circ = 0.5236\,\text{rad}$$

$$T_1 = 1.272 \cdot \left(1 + \frac{1}{16} \cdot (0.1047)^2\right) \approx 1.273\,\mathrm{s}$$
$$T_2 = 1.272 \cdot \left(1 + \frac{1}{16} \cdot (0.5236)^2\right) \approx 1.314\,\mathrm{s}$$

**Time Difference**

$$\Delta T = T_2 - T_1 = 1.314 - 1.273 = \boxed{0.041\,\mathrm{s}}$$

**Oscillations Required for Detection**

To detect a 1-second total time difference:

$$N = \frac{1\,\mathrm{s}}{0.041\,\mathrm{s}} \approx \boxed{25 \text{ oscillations}}$$

Because the difference is more than the precision of the chronometer, 1 oscillation is enough to see the difference.