ABDULLAH GÜL ÜNİVERSİTESİ

PROJECT REPORT

Computation and Analysis (COMA) Capsule

PROJECT NAME: PROJECTILE MOTION AND THE AIR RESISTANCE

PROJECT TEAM

AYŞE ECE AKKURT
HATİCE KÜBRA DURU
SILA DOĞAN

1. INTRODUCTION

OBJECTIVE

The objective of the project is the investigation of projectile motion from different angles. The main purpose is to compare acceleration due to the air resistance and theoretical results.

BACKGROUND

The main structure of the project was discussed, and tasks were shared by the team members. Then, the necessary equipment is decided and collected. After that, the corresponding information that was collected by team members had been decided and combined. Then, to collect data, experiments on projectile motion were done and recorded by a camera. After collecting data, information was analyzed by using "Tracker Simulation" program. Finally, analyzed data used in equations to compare results.

SIMULATION ANALYZE

1)Description of Simulation:

Equipment:

- Spring gun
- Camera
- Meter
- Chronometer

By using the spring gun, a ball is thrown from a top that has a 1-meter height. The point that the ball reached ground had been obtained and the distance was measured between the points where the ball fell and was thrown. This experiment was recorded by the camera and uploaded Tracker program. After that, the video was analyzed by sensitive measurements.

Step by step using simulation process:

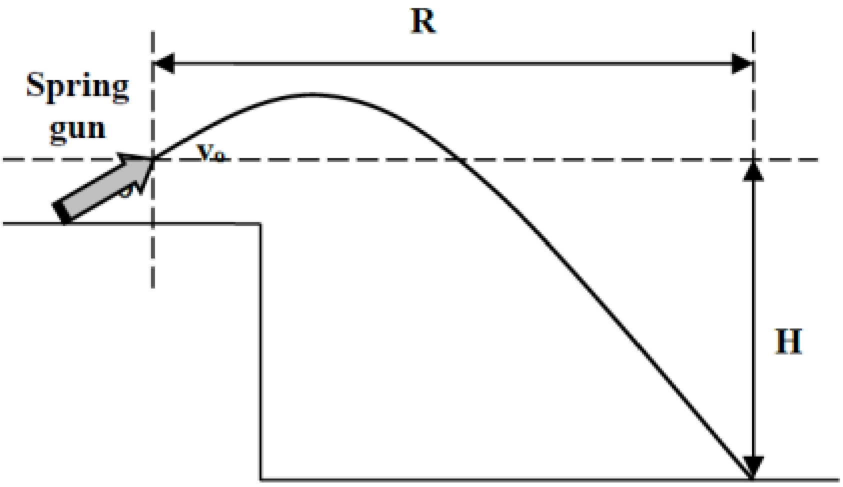
1- The points, where the motion started and ended, were marked.
2- A coordinate system was placed so that the starting point of the movement was the origin.

3- The motion of the ball was marked in small time intervals.
4- A calibration bar and tape measure were added.
After the process, the results were noted and used in calculations by using the equations below:
2)Used Equations:

- $H_{max=}$ $V_0(\sin\theta)^2/2g$: this equation gives the maximum motion height that the ball fell to the level where the motion started.
- $R=V_0.\cos\theta.t$: this equation gives the distance that the ball takes in the x-axis.

2. PHYSICS I

2.1 THEORETICAL TASKS:



2.1.1

$\boldsymbol{v_0} \cdot \mathbf{cos}\theta$ and $\boldsymbol{v_0} \cdot \mathbf{sin}\,\boldsymbol{\theta}$ are the components of $\boldsymbol{v_0}$

$\boldsymbol{x} = \boldsymbol{v_0} \cdot \mathbf{cos}\theta \ \cdot \boldsymbol{t}$ (the distance traveling on the x-axis)

$\boldsymbol{t} = \frac{\boldsymbol{x}}{\boldsymbol{v_0} \cdot \mathbf{cos}\,\boldsymbol{\theta}}$ (by using the equation for x, time (t) is found)

$\boldsymbol{y} = \boldsymbol{H} + \boldsymbol{V_0} \cdot \mathbf{sin}\,\boldsymbol{\theta} \cdot \boldsymbol{t} - \frac{\boldsymbol{g} \cdot \boldsymbol{t}^2}{2}$ (H represents the height of the top)

$$\boldsymbol{y} = \boldsymbol{v_0} \cdot \mathbf{sin}\,\boldsymbol{\theta} \cdot \frac{\boldsymbol{x}}{\boldsymbol{v_0} \cdot \mathbf{cos}\,\boldsymbol{\theta}} - \boldsymbol{g}\frac{\boldsymbol{x}^2}{2 \cdot \boldsymbol{v_0}^2 \cdot \mathbf{cos}^2\,\boldsymbol{\theta}} + \boldsymbol{H}$$

$$y = \tan\theta \cdot x - \frac{g \cdot x^2(\tan\theta + 1)}{2 \cdot v_0^2} + H$$

$2v_0^2 \cdot R \cdot \tan\theta - g \cdot R^2(\tan^2\theta + 1) + 2v_0^2 \cdot H = 0$ (by using back-substitution, R is written instead of x.)

$$\tan^2\theta \, . R^2 . g - 2v_0^2 \cdot R \cdot \tan\theta + g \cdot R^2 - 2v_0^2 \cdot H = 0$$

$$\Delta = b^2 - 4ac$$

$a = R^2 \cdot g$ $\quad$ $b = R \cdot V_0^2 \cdot 2$ $\quad$ $c = g.R^2 - H.2Vo^2$

$$\Delta = (2.R.Vo^2)^2 - 4R^2.g(g.R^2 - H.2Vo^2)$$

$$= 4R^2(Vo^4 - R^2.g^2 + 2HVo^2)$$

$$tan\theta = 2.R.Vo^2 \pm \frac{2.R\sqrt{Vo^4 - R^2.g^2 + 2HVo^2}}{2.R^2.g}$$

$$= \frac{1}{R.g}(Vo^2 \pm \sqrt{Vo^4 - R^2.g^2 + 2HVo^2})$$

2.1.2

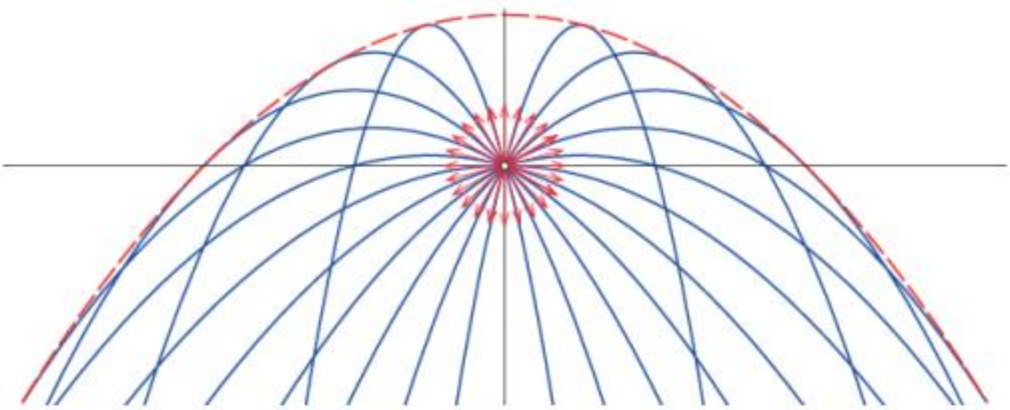


Figure 2. The envelope curve $y = f(x)$ (red dashed line).

$t_{upper\ point} = \frac{V_0 \cdot \sin\theta}{g}$ $\qquad$ $H\max - H = \frac{V_0^2 \cdot \sin^2\theta}{2g}$

$2t_{upper\ point} = \frac{2V_0 \cdot \sin\theta}{g}$ $\qquad$ $r = \frac{V_0^2 \cdot \sin 2\theta}{2g}$ $\qquad$ $x = V_0 \cdot \cos\theta . t$

$$f(x) \to T\left(0, \frac{Vo^2}{2g}\right) \qquad r = \frac{V_0^2}{g} \qquad -r = -\frac{V_0^2}{g}$$

$$f(x) = a.\left(x - \frac{V_0^2}{g}\right).\left(x + \frac{V_0^2}{g}\right) \qquad f(0) = a - \frac{V_0^4}{g^2} = \frac{V_0^2}{2g}$$

$$a = \frac{V_0^2}{2g}.-\frac{g^2}{Vo^4} = -\frac{g}{2\text{Vo}^2}$$

$$f(x) = -\frac{g}{2\text{Vo}^2}.x^2 + \frac{V_0^2}{2g}$$

2.2 EXPERIMENTAL TASKS:

2.2.1 Average Velocity

With the spring gun shooting experiment, different data were collected and analyzed by team members.

ANGLE	0.5 °	32.6 °	46.9 °
V_0(initial velocity)	≈ 4.39	≈ 4.52	≈ 4.672

$$V_{ort} = 4.39+4.52+4.672$$

$$\approx 4.52$$

2.2.2 The difference between reality and theoretical calculations

Projectile motion in real life can be different from theoretical calculations due to several factors, leading to differences between collected data and predictions. Some reasons for the difference between real-life and theoretical calculations:

1- **Air Resistance:** The theoretical calculations for projectile motion generally assumed no air resistance. In reality, air resistance significantly affects the motion of the projectile. This effect can cause the projectile to follow a different trajectory and reduce the range.

2- **Wind:** Wind can have a significant impact on the trajectory of a projectile. The theoretical calculations assume no wind, but in real life, wind can change the path and influence the landing point.

3- **Deviation of Launch Angle:** Theoretical calculations assume perfect launch conditions. In reality, it's challenging to achieve and measure these conditions precisely, without proper context.

4- **Measurement Errors:** Errors in measuring the initial variables, such as the launch angle, initial velocity, and time of flight, can cause differences between theoretical and collected data.

An example that refers to the experiment, below, shows the differences between reality and theoretical calculations:

Degrees	0.5°	32.6°	46.9°
L(experiment)	2.376 m	2.925 m	2.726 m
L(calculations)	1.9763 m	2.862 m	2.8829 m
H(experiment)	1 m	1.463 m	1.69 m
H(calculations)	1.000067 m	1.296 m	1.58 m

*(all data are approximate values)

2.2.3 Air Resistance

The air resistance can significantly affect the projectile motion. The opposite force acts against the object (air resistance) and affects the average acceleration component areas due to constant air resistance in projectile motion. In projectile motion with air resistance, the relevant forces included are gravity and air resistance. Gravity acts vertically downward, while air resistance acts opposite to the direction of the projectile's motion.

The net force can be calculated by using Newton's second law:

$$F_{net}=m.a_{avg}$$

The net force can be calculated as:

$$F_{net=}\ F_{gravity}\text{-}\ F_{air}$$

$F_{gravity}=m.g$

$F_{air=}-\frac{1}{2}.C.\ \rho.A.v^2$ (C is the air resistance coefficient, ρ is the density of air, A is the cross-sectional area of the object, v is the velocity)

$F_{air=} -\frac{1}{2}.(1.225).(0.32).(0.12).(4.5)^2 = 0.47$($\rho$≈1.225 kg/m^3, C≈0.32, A≈0.12)

$F_{gravity}$=(3.8).(9.8)=37.24

$F_{net=}$ (37.24) - (0.47)=36.77

3. CALCULUS I

3.1 Maximum R and H for all possible initial angles θ

$$\boldsymbol{H}\mathbf{max} = \frac{V_0^2 \cdot \sin^2\theta}{2g} + \mathbf{H}$$

$$\boldsymbol{R}\mathbf{max} = \boldsymbol{Vo.cos\theta}\left(\frac{\mathbf{Vo.\sin\theta} + \sqrt{\mathbf{Vo^2.\sin^2\theta + 2gH}}}{\boldsymbol{g}}\right)$$

3.2 Derivatives of df/dx and d 2 f/dx2

$$\boldsymbol{f'(x)} = -\frac{\boldsymbol{g \cdot x}}{\mathbf{Vo^2}}$$

$$\boldsymbol{f''(x)} = -\frac{\boldsymbol{g}}{\mathbf{Vo^2}}$$

$$\boldsymbol{max\,point\ of\ the\ function} = \left(\mathbf{0}, \frac{\mathbf{Vo^2}}{\mathbf{2}\boldsymbol{g}}\right)$$

4. LINEAR ALGEBRA

4.1 Creatin A matrix, inverse of A by Gauss-Jordan, elimination matrices of A, LU factorization of A

$$\boldsymbol{tan\theta} = \frac{\boldsymbol{f(x)}}{\boldsymbol{x}} \qquad \boldsymbol{\theta} = \boldsymbol{arctan}\frac{\boldsymbol{f(x)}}{\boldsymbol{x}}$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 55 & 1 & 0 & 0 \\ 2 & \frac{93}{100} & 25 & 0 & 1 & 0 \\ 3 & \frac{13}{100} & \frac{5}{2} & 0 & 0 & 1 \end{array}\right) \quad \mathrm{R_2 - 2.R_1 \rightarrow R_2}$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 55 & 1 & 0 & 0 \\ 0 & \frac{-187}{100} & -85 & -2 & 1 & 0 \\ 3 & \frac{13}{100} & \frac{5}{2} & 0 & 0 & 1 \end{array}\right) \qquad \mathrm{R_3 - 3.R_1 \rightarrow R_3}$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 55 & 1 & 0 & 0 \\ 0 & \frac{-187}{100} & -85 & -2 & 1 & 0 \\ 0 & \frac{-407}{100} & \frac{-325}{2} & -3 & 0 & 1 \end{array}\right) \quad R_2/(-\frac{187}{100}) \to R_2$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 55 & 1 & 0 & 0 \\ 0 & 1 & \frac{500}{11} & -\frac{200}{187} & -\frac{100}{187} & 0 \\ 0 & \frac{-407}{100} & \frac{-325}{2} & -3 & 0 & 1 \end{array}\right) \quad R_3-(-\frac{407}{100}).R_2 \to R_3$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 55 & 1 & 0 & 0 \\ 0 & 1 & \frac{500}{11} & -\frac{200}{187} & -\frac{100}{187} & 0 \\ 0 & 0 & \frac{45}{2} & \frac{23}{17} & -\frac{37}{17} & 1 \end{array}\right) \quad R_3/(\frac{45}{2}) \to R_3$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 55 & 1 & 0 & 0 \\ 0 & 1 & \frac{500}{11} & -\frac{200}{187} & -\frac{100}{187} & 0 \\ 0 & 0 & 1 & \frac{46}{765} & -\frac{74}{765} & \frac{2}{45} \end{array}\right) \quad R_2-(\frac{500}{11}).R_3 \to R_2$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 55 & 1 & 0 & 0 \\ 0 & 1 & 0 & -\frac{2800}{1683} & \frac{6500}{1683} & -\frac{200}{99} \\ 0 & 0 & 1 & \frac{46}{765} & -\frac{74}{765} & \frac{2}{45} \end{array}\right) \quad R_1-55.R_3 \to R_1$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{7}{5} & 0 & -\frac{353}{153} & \frac{814}{153} & -\frac{22}{9} \\ 0 & 1 & 0 & -\frac{2800}{1683} & \frac{6500}{1683} & -\frac{200}{99} \\ 0 & 0 & 1 & \frac{46}{765} & -\frac{74}{765} & \frac{2}{45} \end{array}\right) \quad R_1-(\frac{7}{5}).R_2 \to R_1$$

$$\left(\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{37}{1683} & -\frac{146}{1683} & \frac{38}{99} \\ 0 & 1 & 0 & -\frac{2800}{1683} & \frac{6500}{1683} & -\frac{200}{99} \\ 0 & 0 & 1 & \frac{46}{765} & -\frac{74}{765} & \frac{2}{45} \end{array}\right)$$

Elimination matricies of A

$$E_{21}=\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad E_{31}=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -3 & 0 & 1 \end{bmatrix} \quad E_{32}=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{407}{100} & 1 \end{bmatrix}$$

$$E_{23}=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & \frac{500}{11}\\ 0 & 0 & 1\end{bmatrix} \quad E_{13}=\begin{bmatrix}1 & 0 & -55\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix} \quad E_{12}=\begin{bmatrix}1 & -\frac{7}{5} & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$A = L \cdot \mathrm{U}$$

$$\begin{bmatrix}1 & \frac{7}{5} & 55\\ 2 & \frac{93}{100} & 25\\ 3 & \frac{13}{100} & \frac{5}{2}\end{bmatrix}=\begin{bmatrix}1 & 0 & 0\\ 2 & 1 & 0\\ 3 & \frac{37}{17} & 1\end{bmatrix}.\begin{bmatrix}1 & \frac{7}{5} & 55\\ 0 & -\frac{187}{100} & -85\\ 0 & 0 & \frac{45}{2}\end{bmatrix}$$

4.2 Linear system which has the constant vector b

A x = b

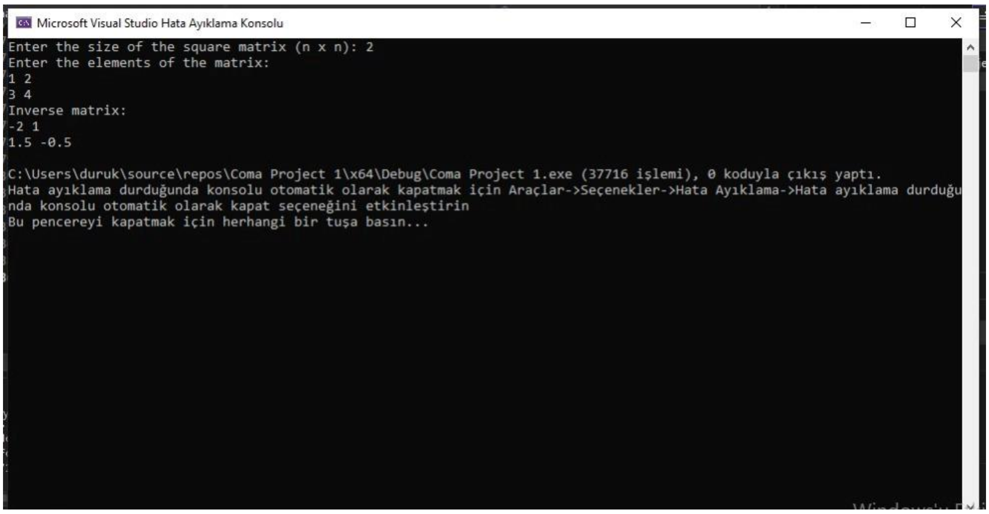
$$A = \begin{bmatrix}1 & 1.40\\ 2 & 0.93\end{bmatrix} \quad x = \begin{bmatrix}x_1\\ x_2\end{bmatrix} \qquad b = \begin{bmatrix}3.52\\ 1.27\end{bmatrix}$$

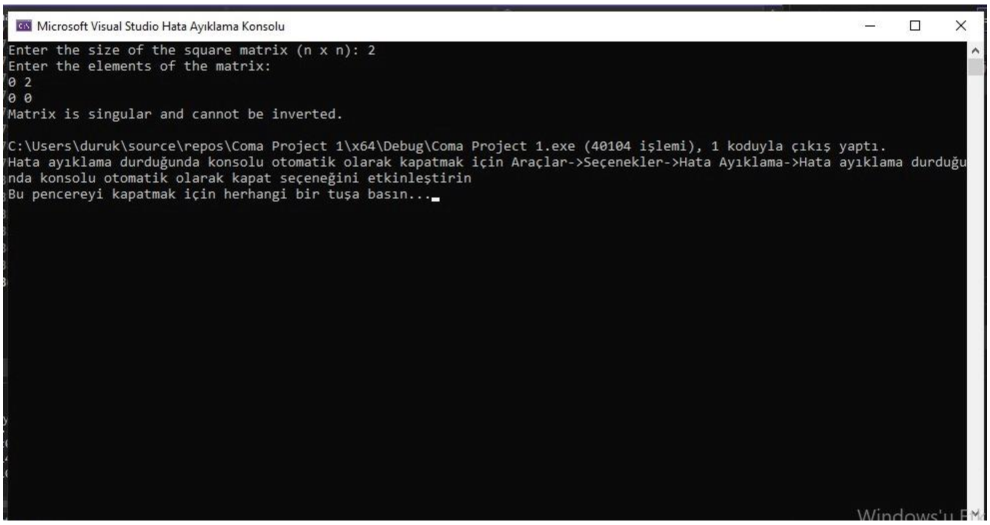
$$x = \begin{bmatrix}-0.79\\ 3.08\end{bmatrix}$$

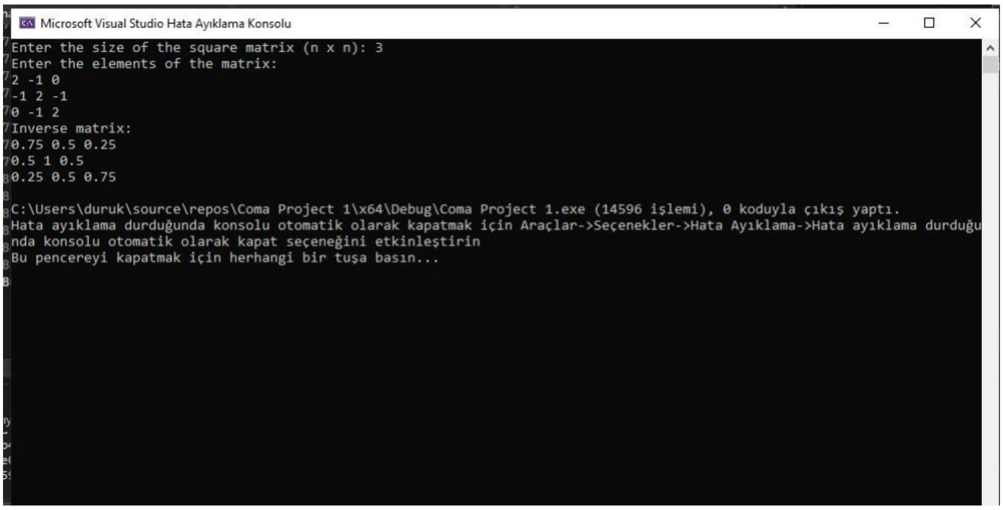
5. INTRODUCTION TO PROGRAMMING

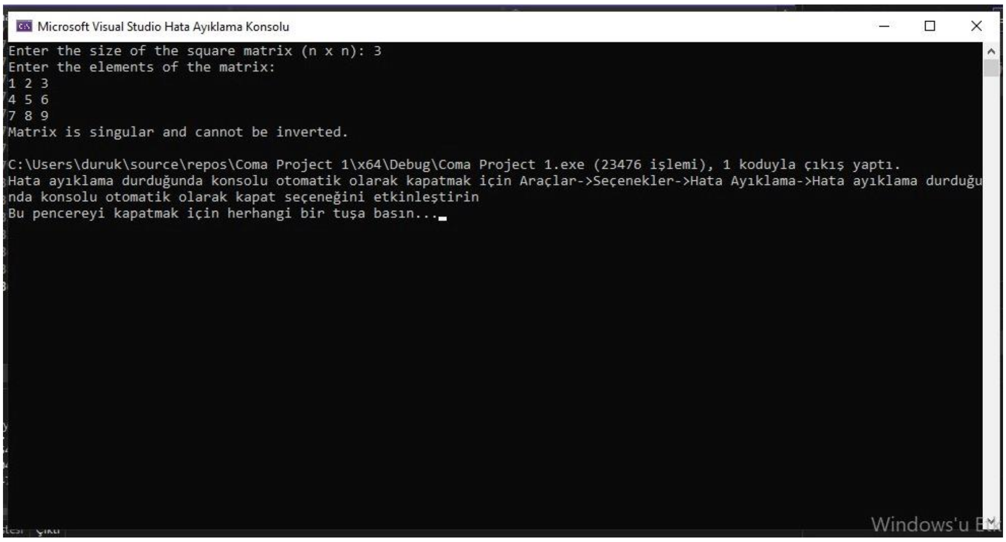
```
#include <iostream>
#include <vector>

using namespace std;

// A function for computing the determinant of a matrix.
double determinant(const vector<vector<double>>& mat, int n) {
    if (n == 1) {
        return mat[0][0];
    }
    else {
        double det = 0;
        for (int i = 0; i < n; i++) {
            vector<vector<double>> submatrix(n - 1, vector<double>(n - 1));
            for (int j = 1; j < n; j++) {
                for (int k = 0, l = 0; k < n; k++) {
                    if (k != i) {
                        submatrix[j - 1][l++] = mat[j][k];
                    }
                }
            }
            det += (i % 2 == 0 ? 1 : -1) * mat[0][i] * determinant(submatrix, n -
1);
        }
        return det;
    }
}

int main() {
    int n;
```

```
    cout << "Enter the size of the square matrix (n x n): ";
    cin >> n;

    // Verify if the size of the input matrix is valid, meaning it is a positive
integer.
    if (n <= 0) {
        cout << "Invalid matrix size. Please enter a positive integer." << endl;
        return 1; // Issue an error code as an indicator of a failure.
    }

    // Create a matrix of dimensions 2n x n that can accommodate both the
original matrix and an identity matrix added as an augmentation
    vector<vector<double>> matrix(n, vector<double>(2 * n));

    cout << "Enter the elements of the matrix:" << endl;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrix[i][j]; // "Obtain matrix values or entries from the
user."
        }
        // Assign a value of 1 to the matching element in the augmented identity
matrix.
        matrix[i][n + i] = 1;
    }

    // Determine if the matrix is singular by checking whether its determinant
equals zero.
    if (determinant(matrix, n) == 0) {
        cout << "Matrix is singular and cannot be inverted." << endl;
        return 1; //Provide an error code to signal an unsuccessful outcome.

    // Utilize the Gauss-Jordan elimination method to compute the inverse.
    for (int i = 0; i < n; i++) {
        double pivot = matrix[i][i];
        // Normalize the entire row by dividing it by the pivot element to
achieve a value of 1
        for (int j = 0; j < 2 * n; j++) {
            matrix[i][j] /= pivot;
        }
        // Subtract the pivot row from the remaining rows to ensure that elements
both below and above the pivot become zero.
        for (int k = 0; k < n; k++) {
            if (k != i) {
                double factor = matrix[k][i];
                for (int j = 0; j < 2 * n; j++) {
                    matrix[k][j] -= factor * matrix[i][j];
                }
            }
        }
    }

    // Output the inverse matrix.
    cout << "Inverse matrix:" << endl;
    for (int i = 0; i < n; i++) {
        for (int j = n; j < 2 * n; j++) {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }

    return 0; // Provide a return value of 0 to signify a successful completion.
}
```


Microsoft Visual Studio Hata Ayıklama Konsolu
Enter the size of the square matrix (n x n): 2
Enter the elements of the matrix:
1 2
3 4
Inverse matrix:
-2 1
1.5 -0.5
C:\Users\duruk\source\repos\Coma Project 1\x64\Debug\Coma Project 1.exe (37716 işlemi), 0 koduyla çıkış yaptı.
Hata ayıklama durduğunda konsolu otomatik olarak kapatmak için Araçlar->Seçenekler->Hata Ayıklama->Hata ayıklama durduğunda konsolu otomatik olarak kapat seçeneğini etkinleştirin
Bu pencereyi kapatmak için herhangi bir tuşa basın...


Microsoft Visual Studio Hata Ayıklama Konsolu
Enter the size of the square matrix (n x n): 2
Enter the elements of the matrix:
0 2
0 0
Matrix is singular and cannot be inverted.
C:\Users\duruk\source\repos\Coma Project 1\x64\Debug\Coma Project 1.exe (40104 işlemi), 1 koduyla çıkış yaptı.
Hata ayıklama durduğunda konsolu otomatik olarak kapatmak için Araçlar->Seçenekler->Hata Ayıklama->Hata ayıklama durduğunda konsolu otomatik olarak kapat seçeneğini etkinleştirin
Bu pencereyi kapatmak için herhangi bir tuşa basın...


Microsoft Visual Studio Hata Ayıklama Konsolu
Enter the size of the square matrix (n x n): 3
Enter the elements of the matrix:
2 -1 0
-1 2 -1
0 -1 2
Inverse matrix:
0.75 0.5 0.25
0.5 1 0.5
0.25 0.5 0.75
C:\Users\duruk\source\repos\Coma Project 1\x64\Debug\Coma Project 1.exe (14596 işlemi), 0 koduyla çıkış yaptı.
Hata ayıklama durduğunda konsolu otomatik olarak kapatmak için Araçlar->Seçenekler->Hata Ayıklama->Hata ayıklama durduğunda konsolu otomatik olarak kapat seçeneğini etkinleştirin
Bu pencereyi kapatmak için herhangi bir tuşa basın...



6. CONCLUSIONS

The main objective of the project is to analyze the difference between real-life and theoretical projectile motion by using experiments, measurements, and theoretical calculations. As a result of this analysis, the relation between variables such as range, initial velocity, and height can be realized. The purpose of theoretical calculations is to create mathematical formulas and understand the conditions to apply them in real life. The purpose of experiments with the spring gun is to observe the movement of the ball and to confirm the theoretical data. With experiments, data under different conditions are compared to real-life and theoretical predictions and confirm the accuracy of data. Creating mathematical equations for projectile motion requires several operations such as differentiation, algebra, and matrix inversion. These tasks provide students to improve their knowledge and skills, gain the ability to make mathematical operations, and understand the practicability of motion in real life.