Parabolic Motion Guide

Problem

Grogu has a problem controlling the force, He is trying to control the route of the rock he is launching, but he doesn't know the exact position of the rock over time.

Write a program that calculates the projectile's position in the XY plane from the parabolic motion at different time steps.

Input

The inputs for this program are:

- V_0 - Initial velocity.
- θ - Initial angle.
- y_0 - Initial height.
- t_0 - Initial time
- δ - Time Step Size
- n - Number of steps.

Output

The projectile's trajectorie must be stored in a data structure with the following fields:

- ID
- time
- x
- y
- vx
- vy

They will be stored in an array like the following:

```
[
    { // First Element
        "ID": 'IDA',
        "time": 0,
        "x": 0,
        "y": 30,
        "vx": 2.92317,
        "vy": 8.51206
    },

    ...,

    { // Last Element
```

```
        "ID": 'IDF',
        "time": 0.5,
        "x": 1.46158,
        "y": 33.031,
        "vx": 2.92317,
        "vy": 3.61206
    }
]
```

The maximum number of elements in the array is 25.

Decomposing

If you are wondering how to start, don't worry in the following lines we will provide all you need to start.

First of all, we know how to store a variable:

```
int n = 10;
```

But now, we are not going to save only one value, the main purpose of this exercise is to let you work with data structures.

C has is own reserved word *struct*, that is used to represent a record of something, let's say a book:

```
// Defining Books
struct Books {
    char title[50];
    char author[50];
    char subject[100];
};

int main () {
    // Using them as a type of variable.
    struct Books HarryPotter;

    HarryPotter.title = "Harry Potter";
    HarryPotter.author = "J. K. Rowling";
    HarryPotter.subject = "Fantasy";
}
```

As you can see in the example above, it's just the definition of many data types into a single one.

Now, we can talk about the problem. **Projectile motion** is the *motion* of an object thrown or projected into the air, subject to only the acceleration of gravity. The object is called a *projectile*, and his path is called *trajectory*.

The gravity constant is defined by $9.80665 m/s^2$

{Insertar Imagen sobre tiro parabolico}

We will not deepen in projectile motion, so we will provide you all the required formulas.

Name	Formula
Velocity in X	$V_x = V_0 cos(\theta)$
Velocity in Y	$V_y = V_0 sin(\theta) - gt$
Position in X	$x = V_0 cos(\theta) t$
Position in Y	$y = V_0 sin(\theta) t - (1/2) g t^2$ + h

Defining the variables:

- V_0 - Initial Velocity
- θ - Initial angle
- g - Gravity constant
- t - Time
- h - Initial Height

This problem, as you can see is not something that requires a lot of complexity. Most of the formulas can be coded easily. Maybe you are wondering if you need to program $sin()$ or $cos()$ but, no, you can use *math.h*

```c
#include <math.h>

int main() {
    // Input is in radian
    double result = cos(30);
}
```