Movimiento Parabólico

Juan Rudas, Wilmer Cantillo, Andrés Balseiro

Algoritmia y programación

Programa de Ingenierías, Universidad del Norte

Resumen

En este informe se presenta una descripción completa de la realización total del código que simule el movimiento parabólico con sus respectivas leyes físicas, para esto se tomaron programas como eclipse e IDLE (IDEs de Python) y una gran variedad de librerías entre ellas las más importante Visual Python. Para el método utilizado y los resultados con las consecuencias presentadas en el proceso del conocimiento práctico, también se hará un respectivo estudio. Lo cual se expondrá detalladamente en la sustentación del trabajo. Cabe destacar que será un trabajo un tanto complicado puesto que es el primero de este tipo en realizarse pero que con el acompañamiento del profesor esta dificultad disminuirá de manera notable.

**Palabras claves:** código, IDE, visual Python, programa, proyecto, física, parábola

1. Introducción

A los estudiantes de ingeniería de la universidad del norte que se encuentran cursando la asignatura de algoritmia y programación se les propuso realizar un código basado en temas fundamentales de física mecánica, en este caso movimiento parabólico simple, producido gracias a Python, dentro de las instalaciones de la universidad con los equipos y beneficios brindados por la misma bajo la constante tutoría y monitoria del profesor asignado en este caso Hernán Enrique Pajaro, en esta ocasión se utilizaron métodos anteriormente explicados por el profesor. Aparece por ende la descripción total de los pasos utilizados, descritos de manera previa por el profesor.

2. Objetivos

2.1. Objetivo general

Hacer el respectivo código que permita la simulación del movimiento parabólico utilizando como medio VPYTHON para que de manera directa nos permita desarrollar conocimientos en cuanto a la planificación y diseño para mejorar la técnica en este tipo de procedimientos para las puestas en práctica posteriormente en nuestro contexto académico o bien sea en lugares con mayor dificultad como en el campo laboral

2.2. Objetivos específicos

* lograr construir un mapa de ideas de lo se quiere hacer
* Lograr construir información clara sobre todos los temas físicos implicados.
* Ser capaz de representar gráficamente lo que se desea.
* Desarrollar capacidad de liderazgo y de buena comunicación entre compañeros
* Realizar un pseudocódigo del problema con la intención de plasmarlo mas adelante en el programa
* Determinar los problemas y buscar métodos que los puedan opacar con la ayuda y coordinación del profesor
* Realizar completamente el código y perfeccionar aspectos de calidad
* Realizar la sustentación de todo lo relacionado con el trabajo y tener la capacidad de entender todos los procedimientos utilizados en conjunto con las leyes físicas que entran en este fenómeno

3. Marco conceptual

La programación informática, acortada como programación, es el proceso de diseñar, codificar, depurar y mantener el código fuente de programas computacionales. El código fuente es escrito en un lenguaje de programación. El propósito de la programación es crear programas que exhiban un comportamiento deseado. El proceso de escribir código requiere frecuentemente conocimientos en varias áreas distintas, además del dominio del lenguaje a utilizar, en este caso en general se desea realizar un código que permita simular el fenómeno ocurrido en el movimiento parabólico en el cual van mezclados conocimientos previos, trasmitidos por el profesor y por las distintas áreas de la ciencia como el cálculo y la física las cuales son esenciales para la realización del proyecto.

Errores y dificultades en el proceso:

Durante el proceso de realización del proyecto se presentaron diversas dificultades las cuales se fueron superando, las cuales fueron:

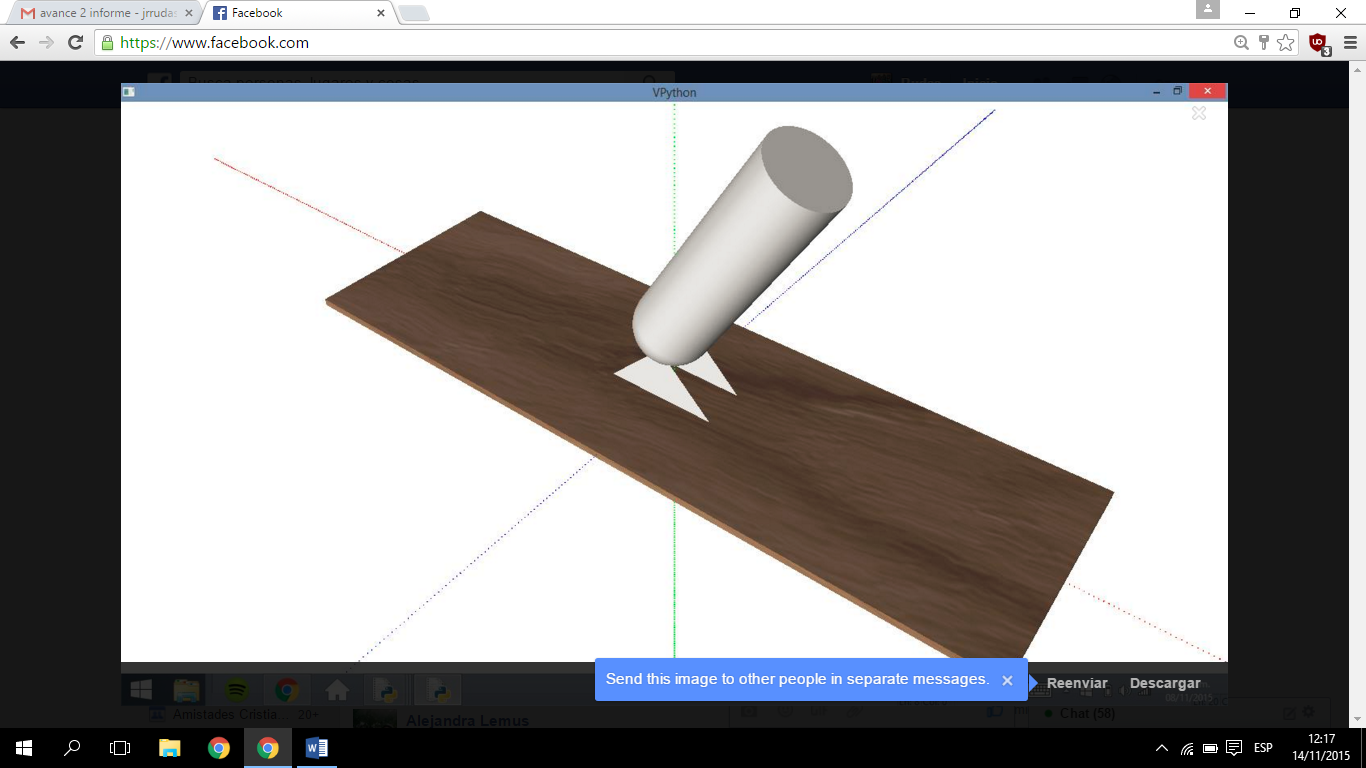
**Realización del cañón:**

Se necesitaba la simulación del objeto expulsado y que esté por consecuencia de dicho impulso tuviera un comportamiento de movimiento parabólico, el grupo luego de discutir decidió hacerlo utilizando una esfera la cuál iba hacer disparada con un cañón. Cabe destacar que también se tuvieron ideas como utilizar un arco, una flecha o la misma esfera, pero en ves de utilizar el cañón se hubiese optado por una catapulta.

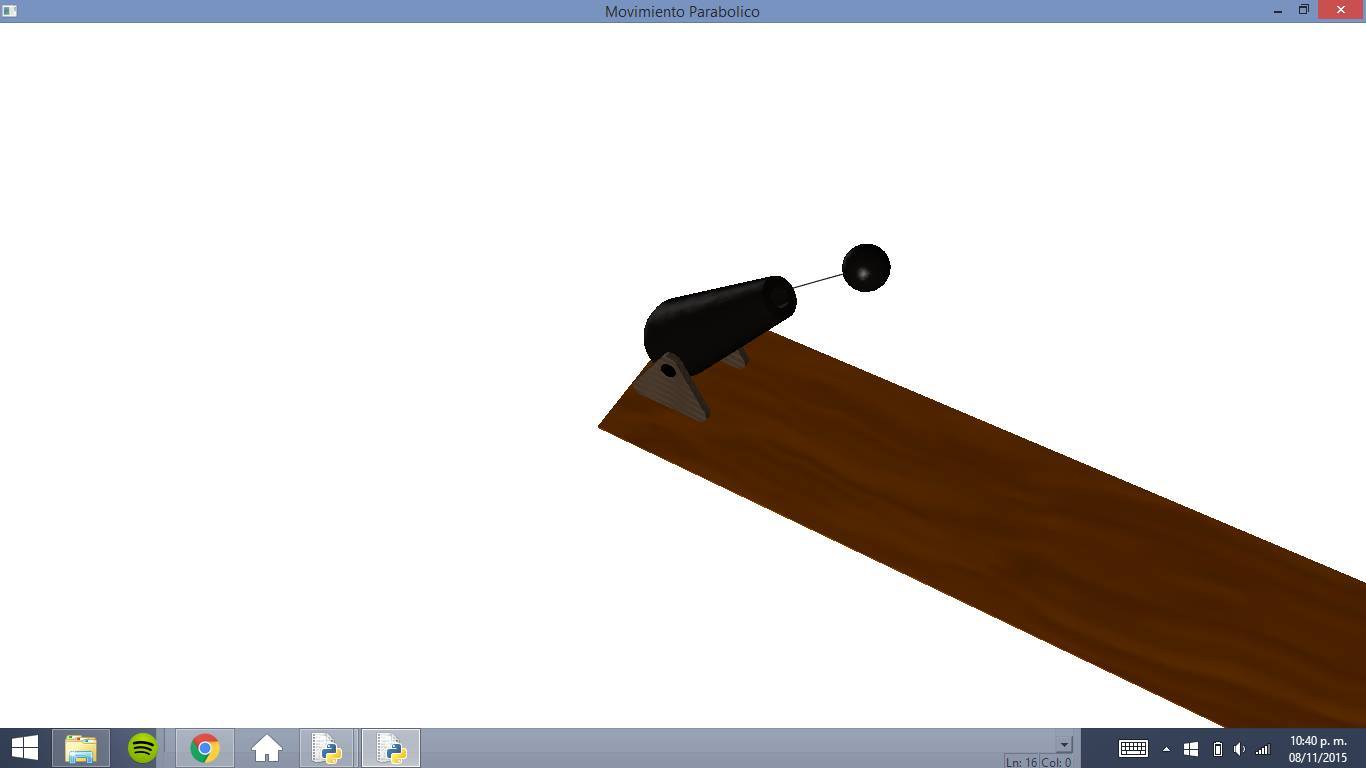
En el momento de hacer el cañón se vieron varios factores que interrumpieron de manera negativa el proceso, las cuales diremos a continuación:

**Diseño:** se presentaron problemas en el diseño y producción del cañón , se hizo una indagación con el profesor asignado sobre si se podían exportar imágenes desde otros programas de diseños 3d como solidwords y AutoCAD, puesto que nos sería más fácil trabajar ese modelado ya que habíamos trabajado con ellos anteriormente, luego de investigar en la página de Python y con algunos ejemplos suministrados por el docente y después de hacer varios ensayos pudimos llegar a nuestro objetivo el procedimiento será explicado a continuación.

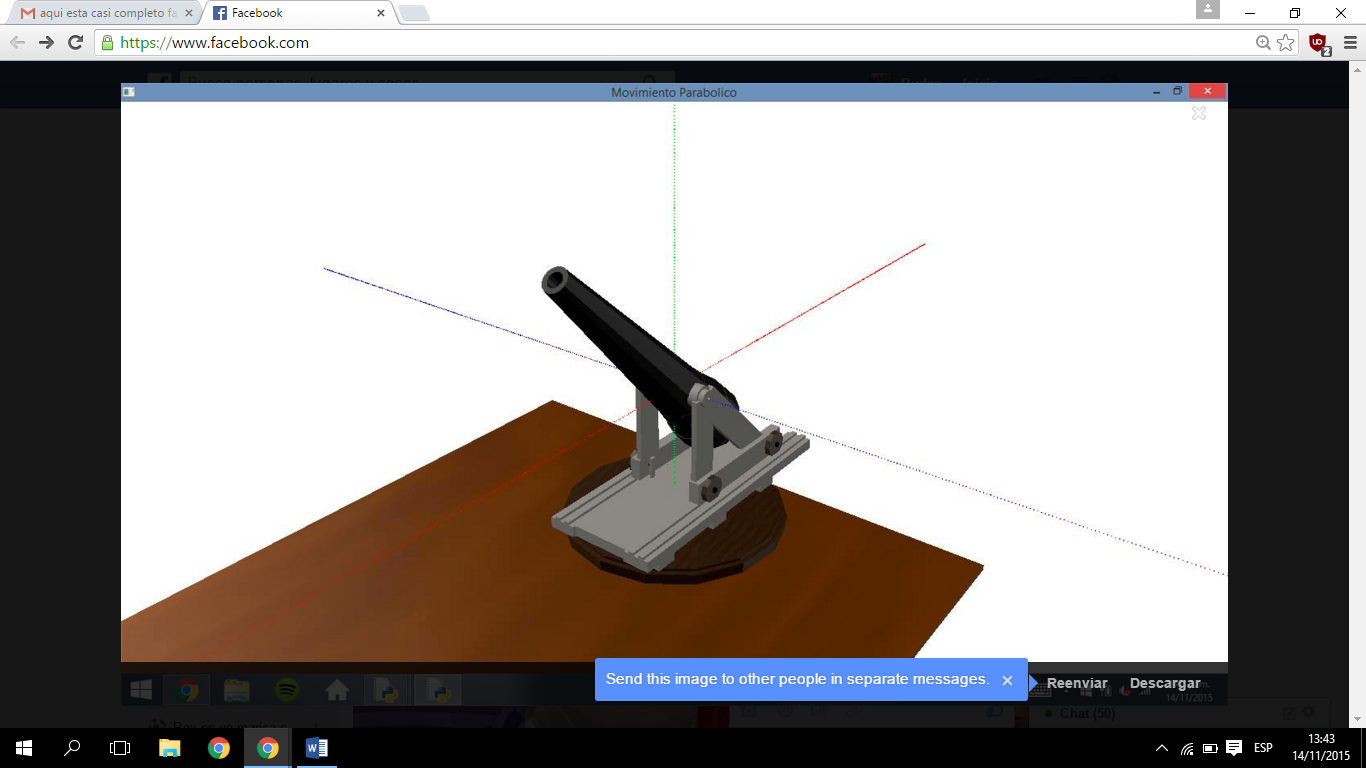
* Se utilizó la herramienta faces para dos triángulos inferiores que serían los que cumplirían el papel de soporte del cañón, y para el cañón se usó la herramienta cylinder la cual consiste en determinar el eje en el cual va a crecer y el radio correspondiente. Lo obtenido se puede apreciar en la **fig.1**
* Para rotarlo se utilizó la función rotate en el cual el ángulo deseado debe ir en radianes y con esto se podría conseguir la inclinación deseada del cañón.

**fig.1**

* Luego de hacer ligeras modificaciones en los colores y un poco en el estilo del cañón en combinación con las dimensiones se logró obtener un prototipo más cercano a los objetivos propuestos lo cual podemos apreciar en la **fig.2**

**fig.2**

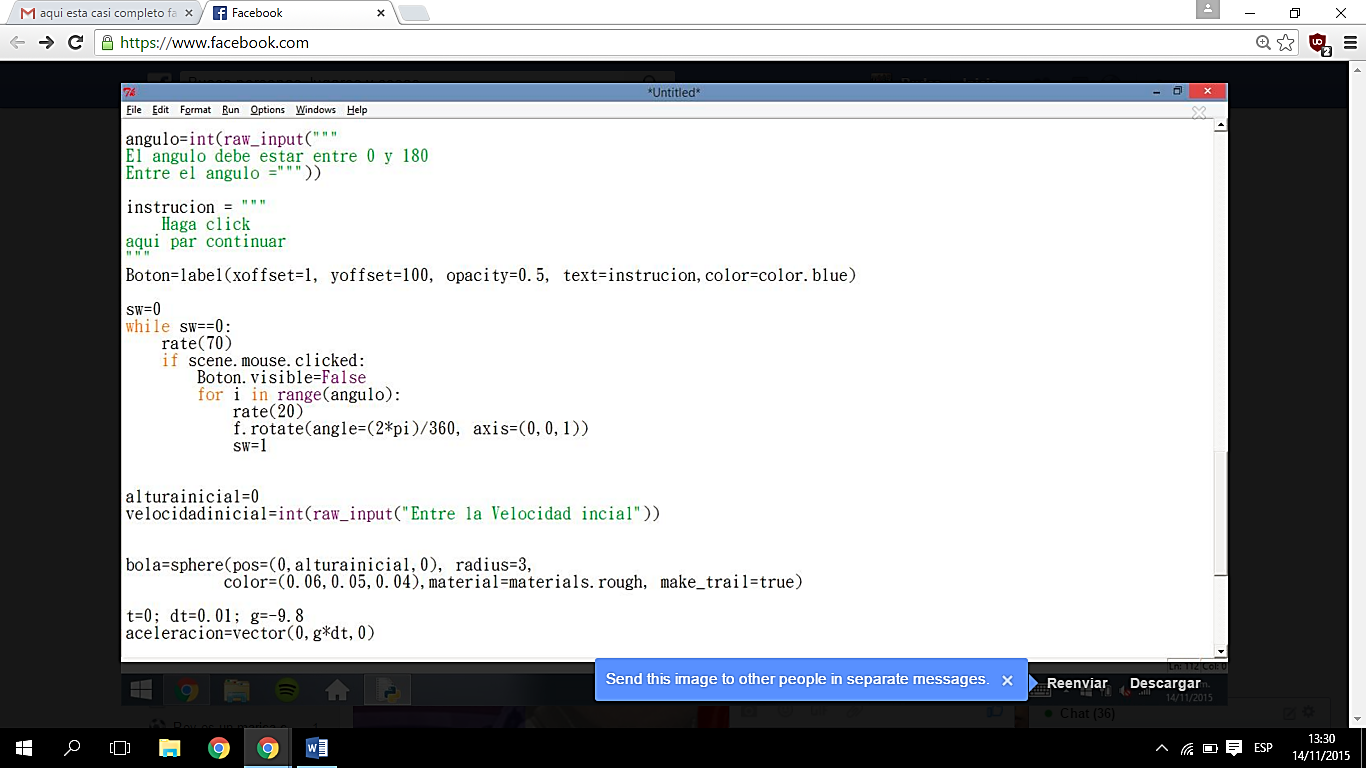
* Por ultimo logramos un diseño bastante realista después que logramos perfeccionar por decirlo de cierta manera las practicas señaladas anteriormente lo anterior se puede apreciar en la **fig.3**

**fig.3**

**Rotación del cañón:** después de solucionar el modelado del cañón, la dificultad que se nos presento fue hacer que el cañón se moviera en el eje Z para lograr el ángulo de inclinación deseado para el disparo del proyectil. Hay que tener encuentra que este ángulo debe ser ingresado por el usuario.

Nos dimos cuenta que con “rotate” podíamos lograr lo deseado pero que esta función presentaba problemas al leer los ángulos ingresados, después de hacer respectivas investigaciones y hacer varios ensayos y consultas con el profesor llegamos a la conclusión que la única manera que teníamos para que el código leyera los ángulos era ingresándolos en modo de radian. Lo cual nos obligó hacer un pequeño algoritmo que convirtiera los ángulos en radianes lo cual facilitaría al usuario.

**Movimiento en tiempo real:** lo que se quería era que el cañón hiciera el movimiento de la inclinación angular deseada en tiempo real, para favorecer el efecto realista exigido por el docente. Lo que se utilizó para lograr esto fue instrucción con el objeto “label” lo cual es un cuadro de texto lo que permite al usuario correr el programa con solo un clic. Lo anterior lo podemos visualizar en la **fig.4**

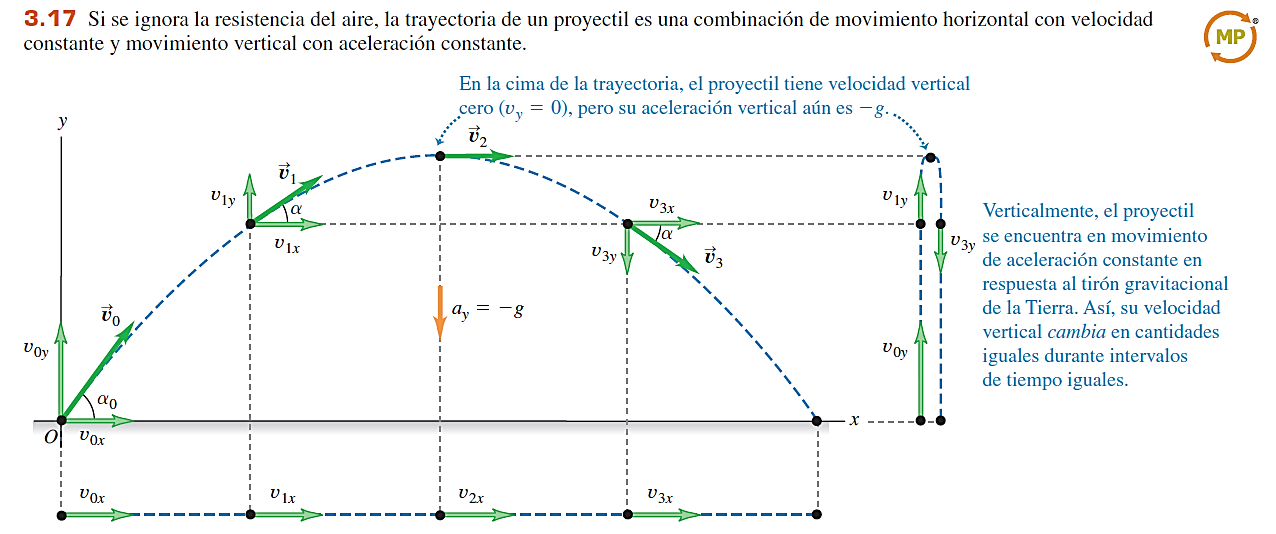
**fig.4**

**Fenómeno físico implicado: “lanzamiento de proyectiles”**

Un proyectil es un cuerpo que recibe una velocidad inicial y luego sigue una trayectoria determinada completamente por los efectos de la aceleración gravitacional y la resistencia del aire. Una pelota bateada, un balón de fútbol lanzado, un paquete que se deja caer desde un avión y una bala disparada por un rifle son proyectiles. El camino que sigue un proyectil se conoce como su trayectoria. Para analizar este tipo de movimiento tan común, partiremos de un modelo idealizado que representa el proyectil como una partícula con aceleración constante (debida a la gravedad) tanto en magnitud como en dirección. Se ignoran los efectos de la resistencia del aire, así como la curvatura y rotación de la Tierra. Como todos los modelos, este tiene limitaciones. La curvatura de la Tierra debe considerarse en el vuelo de misiles de largo alcance; asimismo, la resistencia del aire es de importancia vital para un paracaidista. No obstante, podemos aprender mucho analizando este modelo sencillo. El movimiento de un proyectil siempre se limita a un plano vertical, determinado por la dirección de la velocidad inicial.

Esto se debe a que la aceleración causada por la gravedad es exclusivamente vertical; la gravedad no puede acelerar al proyectil de forma lateral. Por lo tanto, este movimiento es bidimensional. La componente x de la aceleración es cero, y la componente y es constante e igual a -g. (Por definición, g siempre es positiva, pero por las direcciones de coordenadas elegidas es negativa).

Entonces, podemos analizar el movimiento de un proyectil como una combinación de movimiento horizontal con velocidad constante y movimiento vertical con aceleración constante. Lo antes explicado se podrá apreciar en la **fig.5**

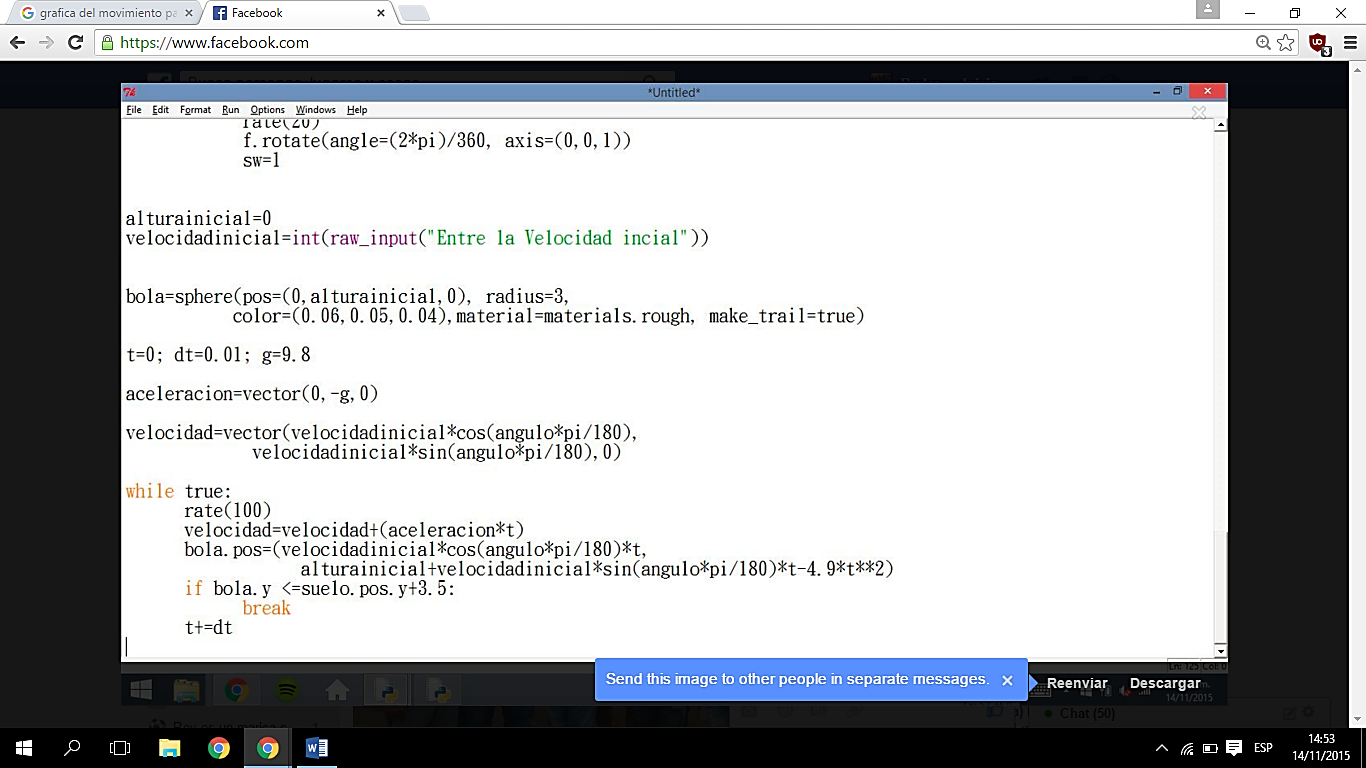
**fig.6**(Tomada del libro Fisica universitaria de de Young Freedman)

**Movimiento parabólico**

A continuación, se presenta la base del movimiento la cual nos permite realizar la animación parabólica

* La parte de altura inicial se deja definida en cero puesto que por decisión se quiso dejar en las coordenadas (0,0,0)
* Por petición del profesor se necesita leer la velocidad inicial lo cual como sabemos se hace con (raw\_input)
* La bola (esfera) se encuentra en la posición (0, alturainicial,0) con un radio de 3 unidades, se eligió un material rustico para favorecer la visual con un color negro (RGB (0.06,0.05,0.04)).
* En el momento de disparar la esfera esta traza una línea la cual resalta o muestra la trayectoria de la misma lo cual es posible con “make\_trail”, una característica de los objetos en vpython.
* El tiempo se inicia en cero y se deja la gravedad indicada en 9.8, con dt siendo el cambio en el tiempo de 0.01.
* La aceleración es un vector que esta dado por las coordenadas (0,-g,0).
* La velocidad está dada por un vector en el cual se incluyen los fenómenos físicos en conjunto con las funciones seno y coseno
* El movimiento de la esfera está dado por un siclo “while true” con un rate =100 siendo este último el que permite graduar la rapidez de como aparezca el movimiento en pantalla lo cual favorece la ilusión de realismo del fenómeno, hay que tener en cuenta que el rate permite graduar la velocidad de las foto-imágenes por segundo de la visual.
* También hay que tener en cuenta que se programó de tal manera que cuando la posición de la pelota sea igual a la posición del piso se saldrá del while true por medio de un break utilizado y así de manera instantánea la pelota dejara de moverse.

El proceso anterior se puede apreciar de manera completa a continuación en la **fig.6**



**fig.6**

# **Bibliografía**

edición+, U. P. (s.f.).

Hewitt, P. G. (2007). *Fiísica Conceptual .* Mexico: Pearson.

YOUNG, H. D. (2013). *Física universitaria volumen 1.* México: PEARSON.