Lektion 5 : Anwenden von ti_system & ti_plotlib | **Anwendung : Der freie Fall**

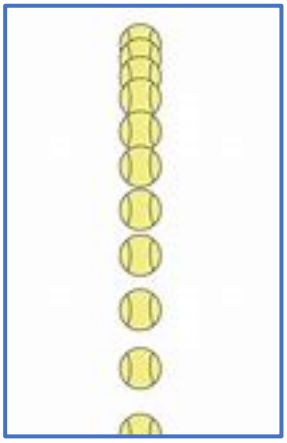
In dieser Anwendung von Lektion 5 werden die in den Lektionen 4 und 5 vorgestellten Befehle verwendet, um eine Simulation zur Beschreibung einer Bewegung zu programmieren.

Lernziele :

- Simulation einer Chronophotographie.
- Übertragen der errechneten Werte in Listen auf dem Taschenrechner.

Mittels der Sprache Python soll ein physikalischer Vorgang simuliert werden. Es handelt sich dabei um den freien Fall :

- Das Programm beschreibt die Bewegung.
- Die grafische Darstellung der errechneten Werte soll in Form einer Chronophotografie erfolgen.
- Schließlich sollen die Werte in Listen exportiert werden..



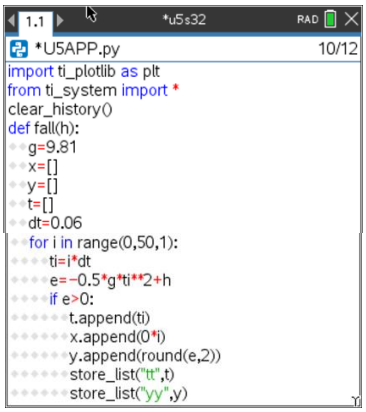
Die Aufgabe : Ein Ball wird aus der Höhe h_0 ohne Anfangsgeschwindigkeit fallen gelassen. Alle 60 ms wird der Ball photografiert.

Es soll ein Programm geschrieben werden, das diesen Vorgang simuliert. Die Ausgabe der Werte soll in einer Grafik erfolgen, die einer Chronophotografie ähnelt.

a) Berechnung der Ballpositionen

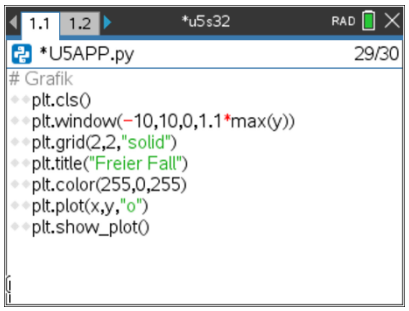
Für den freien Fall aus einer Höhe h_0 ohne Anfangsgeschwindigkeit gilt $y = -\frac{g}{2} \times t^2 + h_0$ mit y>0 und g = 9,81 m/s².

- Ein neues Programm mit dem Namen **U5APP** anlegen.
- Die Module **ti_system** und **ti_plotlib** einbinden.
- Den Bildschirm löschen.
- Eine Funktion **fall(h)** erstellen, die als Parameter die Fallhöhe **h** (=h_0) enthält und die Position des Balles in Abhängigkeit von der Zeit berechnet.
- Drei Listen **x**, **y** und **t** anlegen**.**
- Die y-Werte werden im Abstand von jeweils 60 ms berechnet.
- Die Werte werden in Listen gespeichert, solange y>0 ist.



Verwendet wird eine **For**-Schleife. Die errechneten Höhen **y** werden auf 2 Stellen hinter dem Komma gerundet. Für die grafische Darstellung benötigt man noch die Liste der **x**-Werte, die nur aus Nullen besteht, da der Fall senkrecht nach unten erfolgt. Zum Schluss werden die Listen **t** und **y** exportiert in die Listen **tt** und **yy** des Taschenrechners.

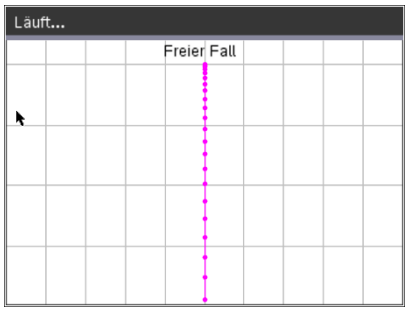
b) Grafische Darstellung



Grafikeinstellungen :

- Grafikbildschirm löschen **plt.cls()**.
- Raster mit dem Abstand 2 erzeugen **plt.grid(xscl, yscl, type,(r,g,b))**.
- Grafikfenster einstellen **plt.window(x_{min}, x_{max}, y_{min}, y_{max})**.
- Farbe festlegen (Magenta) **plt.color(255,0,255)**.
- Die Punktwolke festlegen **plt.plot(x-list, y-list, « mark »)**.
- Den Graphen anzeigen **plt.show_plot()**.

Nun kann man das Programm laufen lasssen. Im Beispiel ist die Funktion fall() mit 8 (m Höhe) aufgerufen worden.



Es ergibt sich das nebenstehende Bild.

Lehrertipp : Man kann auch die Geschwindigkeiten des Balles berechnen und durch kleine Pfeile anzeigen lassen.

c) Anzeige der exportierten Daten

Dazu muss man Python verlassen und ein Streudiagramm **s1** in Graphs anlegen, mit dem sich die Listen direkt darstellen lassen. Es ergibt sich *nicht* das Bild einer Chronophotografie, da hier **yy** = h (= y im Plot) in Abhängigkeit von **tt** (= x im Plot) dargestellt wird.