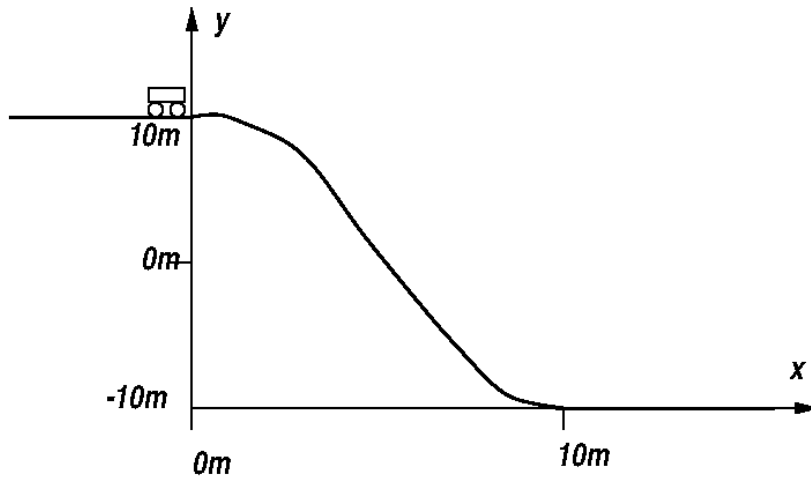


Phys 124 - Freshman Project
Spring 2025
Simulation Homework II



A track has been constructed to transport material from a height of 10m to a lower level at -10m. The horizontal distance between the starting and ending points is also 10m. The carriages will travel from the top of the track to the bottom under the influence of gravity only, so you can take the speed v of the carriage to be affected only by its height. We will have

$$\frac{1}{2}mv^2 = mg(10m - y) \text{ or } v = \sqrt{2g(10m - y)} \quad (1)$$

where y is the y -coordinate of the carriage.

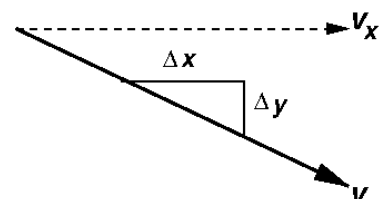
The track was originally designed to fit the curve $y(x) = H \cos(\pi x/L)$ where H and L are both 10m. It is now desired to change this track somewhat so that the carriage will reach the bottom more quickly. You are allowed to modify the curve of the track so that it will take the form

$$y(x) = H \cos(\pi x/L) + A \sin(2\pi x/L)$$

with an A of your choice. (Note that A has to be negative, otherwise the initial slope becomes positive.)

1. Write a python code to simulate the motion of the carriage down the track. You can deduce the speed of the carriage at any time from Eqn. 1 above.

(You will need to find the x -component of the velocity using $v_x = v/\sqrt{1 + (dy/dx)^2}$.)



2. Run the program with the initial condition $x(0) = 0\text{m}$ and $y(0) = 10\text{m}$. You will also need to give a very small initial speed to the carriage so that it starts moving - perhaps $v_x = 0.01\text{m/s}$.
3. Your program should stop when the x -coordinate exceeds 10m and print the value of time at that instant.
4. Run the program with various values of the parameter A to find the path with the shortest travel time.
5. Provide a plot of the x coordinate as a function of time, and a plot of the shape of the track (an $x - y$ plot). How could you change the track to achieve even shorter times?

As an example, my code contains

```
import matplotlib.pyplot as plt
import numpy as np

#####
# Sinusoidal track

plt.clf()

#track parameters:
H = 10.
L = 10.
g = 9.8

dt = 0.0001
n = 100000
x=[]
y=[]
t=[]

A = <this is where you put your guesses for A>

# My initial conditions:
x.append(0.)
y.append(H)
t.append(0.)

# initial push
vx=0.01

for i in range(1,n):

    x.append(<new x value due to vx time dt later>)
    y.append(<y value for that x from y(x)>)
    t.append(i*dt)

    v = <what is v from conservation of energy?>
    dydx = <you will need dy/dx to find the x-component of v>
    vx = <vx from v>

    if x[i] > L : break

print("Total time: ",t[i])

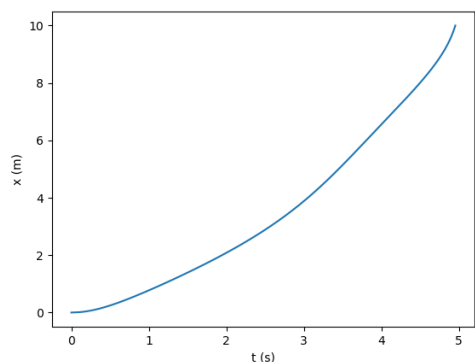
# for plotting x(t)
plt.plot(t,x)
plt.xlabel("t (s)")
```

```
plt.ylabel("x (m)")

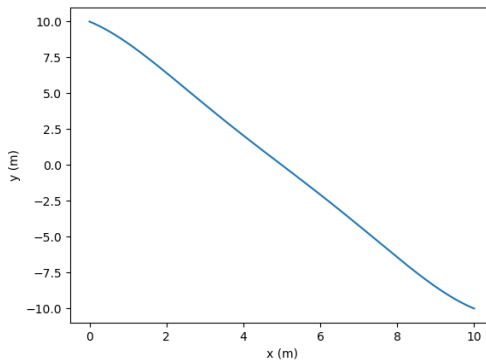
# for plotting y(x)
# plt.plot(x,y)
# plt.xlabel("x (m)")
# plt.ylabel("y (m)")

plt.show()
```

My program gives the plots



x as a function of time



My final $y(x)$

You can check that the size of the time step (Δt) strongly influences the accuracy of the solution.

Please display your results in a PDF manuscript format, using the double-column format of the APS journals and submit it through the Moodle system.

An example LaTeX file is provided through the assignment page to serve as a template which you can modify for your assignment.