MA1008 Python Mini Project report (AY2021/2022, Sem 1)

Project: Cam profile display and follower displacement graph

Name: Liew Yuan Kai

Matriculation no.: U2121420H

Tutorial group: MA2

Tutor: Dr Crivoi Alexandru

1. A guide on how to run the program

Firstly, prepare a text file containing desired inputs. Each line contains information pertinent to the properties of each segment. Below is the format of how to input these variables, within the text file, in the correct order:

(maximum displacement, h)

(base radius, R\_b)

(follower radius,R\_f)

(no. of motion segments, N)

(range of segments, B)

(motion type)

An example of how this is used is shown below:

3.5

5

1

3

90, 90, 180

shm rise, up dwell, shm return

The last two line pertains to the ranges of each segments and each of the segment’s motion types. These inputs are arranged in chronological order. So the first “90” and “shm rise” refers to the properties of the first segments, the second “90” and the “up dwell” refers to the properties of the second segment and so on.

Upon running the program, the user is asked to input the file name with the input prompt “Enter file name:” The user types in the name of the file in the same directory as the program.

After the user hits enter, a message is printed saying “[Drawing... View the windows 'Cam Profile' and 'Follower Displacement graph']”. The two graphics window appears. The program will first draw the cam profile in the window called ‘Cam Profile’ (Fig 1). Subsequently, the follower’s displacement graph is drawn in the window called 'Follower Displacement graph' (Fig 2). These drawings are shown to be drawn in real time.

Once the displacement curve has finished, the program prints “Completed! Check cam profile and displacement graph.” and the program is considered to have stopped.

N.B. The key words for the desired motion types are: shm rise, shm return, cycloidal rise, cycloidal return, constant acceleration rise, constant acceleration return, up dwell, down dwell

1. A picture containing chart

   Description automatically generatedPictures of graphics window:

Fig 1. ‘Cam Profile’ window

Chart, histogram

Description automatically generated

Fig 2. ‘Follower Displacement graph’ window

1. Key strengths and limitations:
2. Strengths:

One overall strength of this program is that it is fairly user friendly. It is visually pleasant, fully labelled and the cam profile is scaled up to have an enlarged view of it. The base circle is drawn in red to distinguish it from the follower and the cam profile.

Another key strength of this written program is that the scale factor of the vertical axis changes dynamically according to the maximum displacement that the user has specified. This is so that a good amount of space is used to display the curve so there is less space wastage of the ‘Follower Displacement graph’ window.

Additionally, error handling is extensive to catch user input errors which is important as many input variables make it highly susceptible for user input errors.

Furthermore, there is a feature to catch unrecognised motion types by using the membership operator in

1. Limitations:

A limitation is that the user must know the correct order of arrangement of the input file prior to running of the program. Another limitation is that the user must know the specific keyword to call the desired motion type.

Also, the ‘dwell’ motion type is subdivided into ‘up dwell’ and ‘down dwell’. A general code that can encompass the situations when the dwell is a zero or non-zero displacement has yet to be found. Up dwell refers to a dwell where the displacement of the follower is positive and non-zero, while down dwell refers to a dwell where the displacement of the follower is 0. While there is thought of using the built-in .ycor() method Turtle method, this can easily solve the issue of the displacement graph, but the cam profile cannot use this method, hence it was not implemented into the code but it was considered as a solution.

Also, in the “Cam Profile” window, there has yet to be a system of code to adjust the scale factor, or translation, of the drawing so that the cam profile can fit into the window. If the user inputs a large maximum displacement, part of the profile may lie outside the window.

By default, Python’s range function only accepts integers as arguments. The step= argument cannot be passed in as a float which would have improved the resolution of the curve, making it seem smoother.