# SYMPY MECHANICS

## Basic Information

**NAME** : Mahi Mishra
**MAJOR** : Computer Science and Engineering
**SPECIALIZATION** : Artificial Intelligence and Machine Learning
**UNIVERSITY** : Lovely Professional University
**GITHUB** : https://github.com/Mahi0411
**LINKEDIN** : https://www.linkedin.com/in/mahi-mishra-ab574a1a6/
**EMAIL** : mahi04mishra@gmail.com
**PHONE** : +917521083261
**ADDRESS** : 1/557 vikas nagar, Lucknow
**TIME ZONE** : Indian Standard Time

## PERSONAL BACKGROUND

I am a Second year undergraduate pursuing B. Tech in Computer Science and Engineering with specialization in Artificial Intelligence and Machine Learning at Lovely Professional University , Phagwara. I have keen interest in Mathematics, Physics and Computer Science and have been following these subjects for quite a long time.
Few of the courses that I have taken in my graduation are calculus, Linear Algebra, Discrete Mathematics, Probability and Statistics, Mechanics oscillations and waves(Some part of classical Mechanics is covered), OOPs and Data Structure and Algorithms.

I started out with C++ in my 10+1 i.e. when I was in my junior college. From writing the basic code structure to implementing OOPs, curiosity kept rising. I built a project in a group using GUI and MY SQL which was based on managing a concert. Now, I've made satisfactory progress in programming to take part in GSOC.

## As a Programmer

I use VS code as my primary source code editor. I am familiar with the workflow of git and github. I have 4 years experience in programming. I have worked in C++ in my 10+1 and 10+2 using code blocks and have done a project using the same as well. Coming in the 1st year of graduation, I have gained a basic knowledge of C and JAVA. I have nearly 2 years of experience in JAVA . The platform I use for coding in JAVA is IntelliJ and VS code. I have created a project making use of JAVA that is a discussion forum.

Python language is one of the most accessible programming languages available because it has simplified syntax and is supported by an extremely large ecosystem of libraries. I became familiar with python in my 2nd semester. As I have worked with other languages previously, studying python was facile. Examining the scope of python in today's world from data visualization, machine learning to many of the leading tech fronts overwhelmed me and I continued to expand my learning. I am working on libraries like Numpy , pandas and Matplotlib and so on. The advanced things too can be worked out on an easy interface. I am engaged in working in Machine Learning so I would say I am quite good at working with python libraries like Numpy, Pandas, Tkinter .

One of my favorite features of sympy is sympify

```
str_expr = "x**2 + 3*x - 1/2"
expr = sympify(str_expr)
expr
x**2 + 3*x - 1/2
```

```
expr.subs(x, 2)
```

## Java Project :

Jan 2021 - Feb 2021

• Make a discussion forum where people can ask questions and can get answers by people on that platform.

• Even upvote other answers and can edit their answers.

• Project was completed in java

**LINK :**

## GUI Calculator:

- A calculator that takes two numbers as input and performs arithmetic operations.
- The result is displayed on the Graphical User Interface(GUI).
- The project is made using python and its library tkinter.

**LINK:**

## Tic-Tac-Toe using Python:

- This game is also known as Noughts and crosses or O's and X's.
- Using the gaming library of python i.e. pygame.

**LINK :**

## Number Guessing game:

- Computer picks a random number and the player makes a guess.
- The guesses are compared and too high, too low or You won is printed

- Project was completed using python library tkinter.

**LINK :**

## Quiz game with python:

- The project is based on a Hogwarts quiz consisting of 20 questions.
- After completion of the quiz a message box pops to tell the marks obtained.
- The project is completed using python library tkinter.

**LINK :**

# MY PROJECT

The aim of my project is to implement specific forces and torques in the Sympy classical mechanics in physics module, so the users can work more efficiently with the complexities of forces and torques.Torque is a measure of the force that can cause an object to rotate about an axis. Just as force is what causes an object to accelerate in linear kinematics, torque is what causes an object to acquire angular acceleration. While working manually with pen and paper we can define variables and create the required equations but with this implementation users can create forces without defining the variables.

The motivation for this project came from the interest in Physics and Python. I think I am quite acquianted with the required theory and I'm keen to learn and improve more throughout the project.

Throughout this project some new submodules like force.py and tor.py would be created under the physics.mechanics module. The force.py would contain classes of various common and typical forces to be used by the user and the tor.py would contain common toques which can be used by the user.

# THE PLAN :

A force is an influence that can change the motion of an object. A force can cause an object with mass to change its velocity (e.g. moving from a state of rest), i.e., to accelerate.

In sympy, Currently under sympy.physics.mechanics , apply_force which apply the force on self or equal and opposite forces on self and other body.

```
from sympy import symbols
>>> from sympy.physics.mechanics import Body,
Point,dynamicsymbols
>>> m, g = symbols('m g')
>>> B = Body('B')
>>> force1 = m*g*B.z
>>> B.apply_force(force1) #Applying force on B's masscenter
>>> B.loads
```

Similarly for torque a function apply_torque is there in sympy.

- ***13th june - 16th june***

  The variables are the key to form an equation. In the sympy module named sympy.physics.mechanics many variables are already declared like Body , Point, etc and functions like loads, apply_torque are already built taking them in consideration we can develop the required variables to implement the different types of forces and torques.

- ***21st june - 17th july***

  After declaring the variables the task is to form the necessary set of equations.

For different types of forces we will first form a separate Abstract class named Force which can be inherited by all other types of forces.  Dealing with the 1st kind of forces i.e LINEAR FORCES  keeping the concept of equilibrium in mind. Directly applying forces from any point on the body by adding the opposite force to the body on that point.

- **SPRINGS AND DAMPER FORCES**

  A spring is an elastic object that stores mechanical energy, after being stretched or compressed they return to their original form. The force is called the spring force. If x is positive (displacement to the right), the resulting force is negative (to the left), and vice versa. In other words, the spring force always acts so as to restore mass back toward its equilibrium position.

  FORMULA  FOR SPRING FORCE IS ,  F = k( x - x0)

  I will work on to form an easy  way to calculate the displacement keeping the required equilibrium condition.

  Similarly, working on all the types of forces such as contact forces, gravitational forces, friction forces, muscle forces and torques. Implementing individual functions to generalize separate equations for the particular type of force, taking in consideration the inertia, the frame of reference and equilibrium . Writing the docstring and testing the theory with the desired test cases respectively for each

type of force. Working out a certain number of examples for all the forces so that anyone using it can grasp it.

- ***18th july - 14 th August***

Symbolic representation of forces will be added along with the mathematical representations to make the code more efficient. Some tutorial examples will be put on so the users can use the build module.  At last the code will be reviewed and the bugs will be fixed and all the required  touch ups will be given.