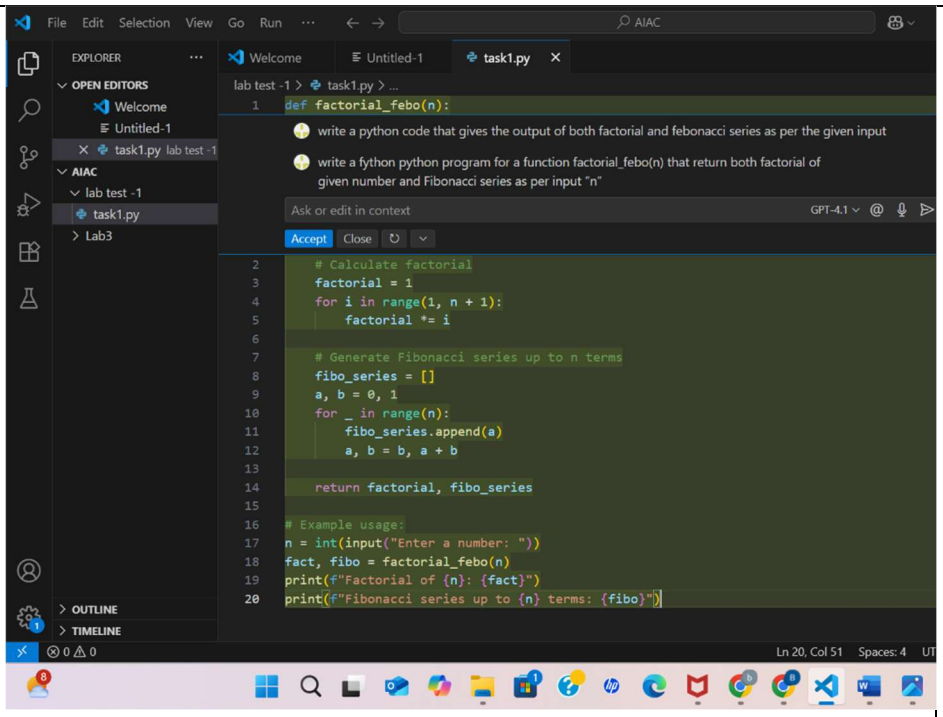
SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
ProgramName:B. Tech		**SET-B**	**AcademicYear:**2025-2026
Roll No.			
Instructor(s)Name		1. Dr. Venkataramana 2. Dr. Ch. Sridhar 3. Mr. Kundan Kumar	
CourseCode	24CS002PC215	**CourseTitle**	AI Assisted Coding
Year/Sem	II/I	**Regulation**	R24
Date and Day of Assignment	22.08.2025	**Time(s)**	01.00PM To 03.00PM
Duration	2 Hours	**Applicableto Batches**	24BTCAIAIB09, 24BTCAIAIB10
Lab Test :1(Present test number)/**4**(Total number of tests)			

Q.No.	Question	*ExpectedTime to complete*
1	**Task Description#1** • Write python program for a function **factorial_febo(n)** that return both factorial of given number and Fibonacci series as per input “n” **Expected Output#1** • **factorial_febo(n) and output** **prompt :** write a python program for a function factorial_febo(n) that return both factorial of given number and Fibonacci series as per input “n” **Code:**	22.08.2025 03.00PM



Output:

Explanation: this code does give the output both factorial and febionacci series of a number

Task Description#2

- Write Python program in cursor AI for **student class** with attributes like Name, Roll no, Marks and also construct a method i.e., **display_details** to display name, rollno, marks and grades

Table: Grades Classification Based on Marks

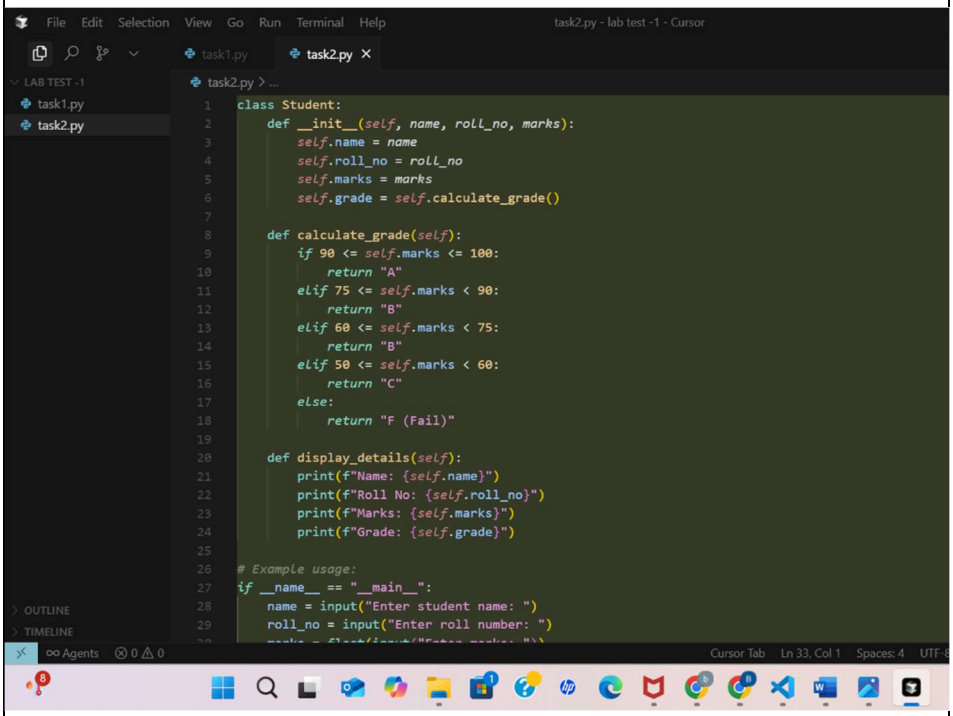
Marks Range (%)	Grade
90 – 100	A+
75 – 89	A
60 – 74	B
50 – 59	C
Below 50	F (Fail)

Expected Output#2

- Student Class and Display Details

- Prompt: write a pyhton code for student class with attributes like Name, Rollno, Marks and also construct a method i.e., display_details to display name, rollno,marks and grades as follows 90 – 100 → A+, 75 – 89 → A,60 – 74 → B,50 – 59 → C,Below 50 → F (Fail)

Code:

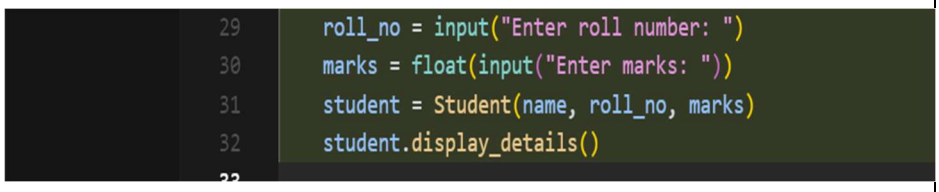



```python
class Student:
    def __init__(self, name, roll_no, marks):
        self.name = name
        self.roll_no = roll_no
        self.marks = marks
        self.grade = self.calculate_grade()

    def calculate_grade(self):
        if 90 <= self.marks <= 100:
            return "A"
        elif 75 <= self.marks < 90:
            return "B"
        elif 60 <= self.marks < 75:
            return "B"
        elif 50 <= self.marks < 60:
            return "C"
        else:
            return "F (Fail)"

    def display_details(self):
        print(f"Name: {self.name}")
        print(f"Roll No: {self.roll_no}")
        print(f"Marks: {self.marks}")
        print(f"Grade: {self.grade}")

# Example usage:
if __name__ == "__main__":
    name = input("Enter student name: ")
    roll_no = input("Enter roll number: ")
```



```python
    roll_no = input("Enter roll number: ")
    marks = float(input("Enter marks: "))
    student = Student(name, roll_no, marks)
    student.display_details()
```

Output:

Explanation: this code is about student marks and grade of the student and details of the student such as name,roll no.

Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output screenshots

Evaluation Criteria:

Criteria	Max Marks
Factorial Function (Task#1)	5
Sorting Function (Task#2)	5
Viva	5
Total	**15 Marks**