1.12. Program to accept marks of the student in five major subjects, calculate the sum, percentage, and find the grade

```
def calculate_grade(percentage):

    if percentage > 85:
        return 'A'
    elif 75 <= percentage <= 85:
        return 'B'
    elif 50 <= percentage < 75:
        return 'C'
    elif 30 <= percentage < 50:
        return 'D'
    else:
        return 'Reappear'

# Accept marks of the student in five major subjects
marks = []
for i in range(1, 6):
    mark = float(input(f"Enter the marks for subject {i}: "))
    marks.append(mark)

# Display the marks
print("Marks obtained in each subject:")
for i, mark in enumerate(marks, start=1):
    print(f"Subject {i}: {mark}")

# Calculate the sum of the marks
total_marks = sum(marks)
print(f"Total Marks: {total_marks}")

# Calculate the percentage
percentage = total_marks / 5
print(f"Percentage: {percentage}%")

# Find the grade
grade = calculate_grade(percentage)
print(f"Grade: {grade}")
```

1.13. Write a program for VIBGYOR Spectrum based on their Wavelength using. Wavelength Range:

```
def find_vibgyor_color(wavelength):
    if 380 <= wavelength < 450:
        return 'Violet'
    elif 450 <= wavelength < 475:
        return 'Indigo'
    elif 475 <= wavelength < 495:
        return 'Blue'
    elif 495 <= wavelength < 570:
        return 'Green'
    elif 570 <= wavelength < 590:
        return 'Yellow'
    elif 590 <= wavelength < 620:
        return 'Orange'
    elif 620 <= wavelength <= 750:
        return 'Red'
    else:
        return 'Wavelength out of VIBGYOR range'

# Accept wavelength from the user
wavelength = float(input("Enter the wavelength (in nm): "))

# Find and display the corresponding VIBGYOR color
color = find_vibgyor_color(wavelength)
print(f"The corresponding VIBGYOR color for wavelength {wavelength}
nm is: {color}")
```

1.14. Write a program for VIBGYOR Spectrum based on their Wavelength using. Wavelength Range:

```
# Constants
G = 6.67430e-11  # Gravitational constant in m^3 kg^-1 s^-2
mass_earth = 5.972e24  # Mass of Earth in kilograms
mass_moon = 7.34767309e22  # Mass of Moon in kilograms
mass_sun = 1.989e30  # Mass of Sun in kilograms
distance_earth_sun = 1.496e11  # Average distance between Earth and
Sun in meters
distance_moon_earth = 3.844e8  # Average distance between Moon and
Earth in meters
# Gravitational force between Earth and Sun
force_earth_sun = G * (mass_earth * mass_sun) / (distance_earth_sun
** 2)
# Gravitational force between Moon and Earth
force_moon_earth = G * (mass_moon * mass_earth) /
(distance_moon_earth ** 2)
# Output the results
print(f"Gravitational force between Earth and Sun:
{force_earth_sun:.2e} N")
print(f"Gravitational force between Moon and Earth:
{force_moon_earth:.2e} N")
# Compare the forces
if force_earth_sun > force_moon_earth:
    print("The gravitational force between Earth and Sun is
stronger.")
else:
    print("The gravitational force between Moon and Earth is
stronger.")
# Explanation
if force_earth_sun > force_moon_earth:
    print("The Earth is more attracted to the Sun than the Moon is
to the Earth.")
else:
    print("The Moon is more attracted to the Earth than the Earth is
to the Sun.")
```