CT-4
DS 1115: Object-Oriented Programming for Data Science
Total Marks: **20**   Duration: 40 Minutes

1. Write a Python generator function called `unique_combinations` that generates all unique combinations of *r* elements from a given list *lst*. The function should take two arguments: *lst* (a list of integers) and *r* (an integer representing the number of elements in each combination). Implement the generator to produce combinations without repeating any element within a combination. **[6]**

   For example: For the list [1, 2, 3, 4], your generator should yield the following combinations:

   (1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)

2. Suppose you have a text file of your favorite book. You're interested in knowing how many unique words the author used in this book. For this, you have to write a Python program to read the text file and count the number of unique words. But the problem is that the book is quite large and reading all the data at once will be very memory intensive. Create a Python generator function that will yield each line of the book at each iteration. Use this generator to count the number of unique words in the book. **[4]**

3. What will be the output of the following snippet of code: **[6]**

```python
import matplotlib.pyplot as plt

x = list(range(-10, 11))

def f(e):

    return e**2

y = [f(i) for i in x]

plt.plot(x, y)

plt.show()
```

4. You are given a CSV file named 'students.csv'. The file contains the following columns: Name, ID, Mathematics, and Science. The Mathematics and Science columns contain the student's marks in mathematics and science respectively. You need to find the student who:

    a. Got the second-highest mark in mathematics.   **[2]**

    b. Got the third-lowest mark in science.   **[2]**