Question 1:

Number game between user and computer. The user starts by entering either 1 or 2 or 3 digits starting from 1 sequentially. The computer can return either 1 or 2 or 3 next digits in sequence, starting from the max number played by the user. User enters the next 1 or 2 or 3 next digits in sequence, starting from the max number played by the computer. Whoever reaches 20 first wins the game.

Note:

- the numbers should be in sequence starting from 1.

- minimum number user or computer should pick is at least 1 digit in sequence

- maximum number user or computer can pick only 3 digits in sequence

```
1 .# number game betwwen user and computer

import random


def computer_turn(current):

    # Computer chooses 1, 2, or 3 sequential numbers, stopping at 20 if possible

    max_choice = min(current + 3, 20)

    choice = list(range(current + 1, max_choice + 1))

    print("Computer Played:", choice)

    return choice[-1]


def user_turn(current):

    while True:

        user_input = input("You Played (enter 1, 2, or 3 sequential numbers starting from {}): 
".format(current + 1))
```

```
        user_numbers = list(map(int, user_input.split(',')))


        # Check if user's numbers are valid
        if len(user_numbers) < 1 or len(user_numbers) > 3:
            print("Invalid input. Enter 1, 2, or 3 numbers.")
            continue
        if user_numbers[0] != current + 1 or any(user_numbers[i] != user_numbers[i - 1] + 1 for i in range(1, 
len(user_numbers))):
            print("Numbers must be sequential starting from", current + 1)
            continue
        if user_numbers[-1] > 20:
            print("You can't go beyond 20.")
            continue


        print("You Played:", user_numbers)
        return user_numbers[-1]


def play_game():
    current = 0
    print("Welcome to the number game! First to reach 20 wins.")
    while current < 20:
        # User's turn
        current = user_turn(current)
        if current >= 20:
            print("Congratulations! You reached 20 and won the game!")
            break
        # Computer's turn
        current = computer_turn(current)
        if current >= 20:
            print("Computer reached 20 and won the game!")
            break


# Start the game
play_game()
```

output:

```
Welcome to the number game! First to reach 20 wins.

You Played (enter 1, 2, or 3 sequential numbers starting from 1): 1
You Played: [1]
Computer Played: [2, 3, 4]
You Played (enter 1, 2, or 3 sequential numbers starting from 5): 5,6,7
You Played: [5, 6, 7]
Computer Played: [8, 9, 10]
You Played (enter 1, 2, or 3 sequential numbers starting from 11): 11,12
You Played: [11, 12]
Computer Played: [13, 14, 15]
You Played (enter 1, 2, or 3 sequential numbers starting from 16): 16
You Played: [16]
Computer Played: [17, 18, 19]
```

```
You Played (enter 1, 2, or 3 sequential numbers starting from 20): 20

You Played: [20]

Congratulations! You reached 20 and won the game!
```

2.#Print Pascal Triangle for given number of rows

```python
def factorial(num):
    """Calculate the factorial of a number."""
    if num == 0 or num == 1:
        return 1
    result = 1
    for i in range(2, num + 1):
        result *= i
    return result


def ncr(n, r):
    """Calculate the number of combinations of n items taken r at a time."""
    if r > n or r < 0:
        return 0
    return factorial(n) // (factorial(r) * factorial(n - r))


def print_pascals_triangle(rows):
    """Print Pascal Triangle with the given number of rows."""
    for i in range(rows):
        # Print leading spaces for formatting
        print(" " * (rows - i), end='')

        for j in range(i + 1):

            print(ncr(i, j), end=' ')

        print()  # New line after each row


# Main function to execute the program

if __name__ == "__main__":

    num_rows = int(input("Enter the number of rows to Print Pascal Triangle: "))

    print_pascals_triangle(num_rows)
```

output:

```
Enter the number of rows to print Pascal Triangle: 4

    1

   1 1

  1 2 1

 1 3 3 1
```

3.# program to print the   repeated elements with frequency count

```
def count_frequencies(numbers):

    """Count the frequency of each element in the list."""

    frequency = {}
    for number in numbers:
        if number in frequency:
            frequency[number] += 1
        else:
            frequency[number] = 1
    return frequency


def print_frequencies(frequency):
    """Print the frequency of each element."""
    for element, count in frequency.items():
        print(f"Element {element} has come {count} times")


if __name__ == "__main__":
    # Read input from the user
    user_input = input("Enter a list of numbers separated by spaces: ")
    numbers = list(map(int, user_input.split()))
    # Count frequencies and print them
    frequency = count_frequencies(numbers)
    print_frequencies(frequency)
```

output:

```
Enter a list of numbers separated by spaces: 1 3 2 4  5 1 2 3 4 5
Element 1 has come 2 times
Element 3 has come 2 times
Element 2 has come 2 times

Element 4 has come 2 times

Element 5 has come 2 times
```

5.# add two objects of class fraction

```
class Fraction:
    def __init__(self, numerator, denominator):
        if denominator == 0:
            raise ValueError("Denominator cannot be zero")
        self.numerator = numerator
        self.denominator = denominator

    def __add__(self, other):
        if not isinstance(other, Fraction):
            return NotImplemented

        # Find a common denominator
        common_denominator = self.denominator * other.denominator
        new_numerator = (self.numerator * other.denominator) + (other.numerator * self.denominator)
        return Fraction(new_numerator, common_denominator)

    def __str__(self):
        return f"{self.numerator}/{self.denominator}"

    def __repr__(self):
        return f"Fraction({self.numerator}, {self.denominator})"

    def simplify(self):
        from math import gcd
        common_divisor = gcd(self.numerator, self.denominator)
        self.numerator //= common_divisor
        self.denominator //= common_divisor

# Example usage
f1 = Fraction(1, 2)
f2 = Fraction(1, 3)

result = f1 + f2
result.simplify()  # Simplifying the result
print(result)
```

Output: 5/6