Question -A

Write a program to display the cos(x) and tan(x) value where x ranges from 0 to 360 in steps of 15.

```
import math

#  function to display cos(x) and tan(x)

def display_cos_tan():
    print("x          cos(x)         tan(x)")
# Headers for better readability
    for x in range(0, 361, 15):
        cos_x = math.cos(math.radians(x))
# Calculate cos(x)
        if x % 180 == 90:
 # Handle undefined values for tan(x) (90, 270, ...)
            tan_x = "Undefined"
        else:
            tan_x = math.tan(math.radians(x))
 # Calculate tan(x)
        print(f"{x:<10}{cos_x:<15}{tan_x:<15}")
 # Print the results

# Call the function
display_cos_tan()
```

Output :

```
x          cos(x)           tan(x)
0           1.0              0.0
15          0.965926          0.267949
30          0.866025          0.57735
45          0.707107         1.0
60          0.5              1.732051
75          0.258819          3.732051
90          0.0              Undefined
105          -0.258819         -3.732051
120          -0.5             -1.732051
135          -0.707107         -1.0
150          -0.866025         -0.57735
165          -0.965926         -0.267949
180          -1.0             0.0
195          -0.965926         0.267949
210          -0.866025         0.57735
225          -0.707107         1.0
240          -0.5             1.732051
255          -0.258819         3.732051
270          0.0              Undefined
285          0.258819           -3.732051
300          0.5              -1.732051
315          0.707107          -1.0
330          0.866025          -0.57735
345          0.965926          -0.267949
360          1.0              0.0
```

Question -B

Create a Python program that recommends movies based on keyword search. The program should:

1. Read a file containing movie data (e.g., title, genre, year, description).

2. Allow the user to search for movies by keyword (e.g., " sci-fi").

3. Display matching movies with their details.

Requirements:

Define a class Movie Recommender with:

o load_movies(self, file_name) - Reads the movie data file.

search_movies(self, keyword) - Searches for movies matching the keyword.

Use file handling and string processing to implement the logic.

• Handle large datasets efficiently

```
class MovieRecommender:
    def __init__(self):
        self.movies = []

    def load_movies(self, file_name):
        """Reads the movie data file and loads it into the
program."""
        try:
            with open(file_name, "r") as file:
                for line in file:
                    # Assuming each line in the file is: Title,
Genre, Year, Description
                    parts = line.strip().split(", ")
                    if len(parts) == 4:
                        title, genre, year, description = parts
                        self.movies.append({
                            "title": title,
                            "genre": genre,
                            "year": year,
                            "description": description
                        })
        except FileNotFoundError:
            print(f"Error: File '{file_name}' not found.")
        except Exception as e:
            print(f"An error occurred: {e}")

    def search_movies(self, keyword):
        """Searches for movies that match the keyword."""
        keyword = keyword.lower()
        matching_movies = [
            movie for movie in self.movies if
            keyword in movie["title"].lower() or
            keyword in movie["genre"].lower() or
            keyword in movie["description"].lower()
        ]
        return matching_movies
```

```
def display_movies(self, movies):
        """Displays the list of matching movies."""
        if not movies:
            print("No movies found matching the keyword.")
        else:
            for movie in movies:
                print(f"Title: {movie['title']}")
                print(f"Genre: {movie['genre']}")
                print(f"Year: {movie['year']}")
                print(f"Description: {movie['description']}")
                print("-" * 40)

# Example usage
if __name__ == "__main__":
    recommender = MovieRecommender()
    recommender.load_movies("movies.txt")  # Use the
path to your file here

    print("Welcome to the Movie Recommender!")
    while True:
        keyword = input("Enter a keyword to search for
movies (or type 'exit' to quit): ")
        if keyword.lower() == "exit":
            break
        results = recommender.search_movies(keyword)
```

Input:

```
Inception, Sci-Fi, 2010, A thief who steals corporate
secrets through dream-sharing technology.
Titanic, Romance, 1997, A love story set on the ill-fated
RMS Titanic.
The Matrix, Sci-Fi, 1999, A hacker discovers reality is a
simulated world.
Avatar, Sci-Fi, 2009, A marine on an alien planet finds
himself torn between two worlds.
The Notebook, Romance, 2004, A romantic story about
enduring love and sacrifice.
```

Output:

```
Welcome to the Movie Recommender!
Enter a keyword to search for movies (or type 'exit' to quit):
sci-fi
Title: Inception
Genre: Sci-Fi
Year: 2010
Description: A thief who steals corporate secrets through
dream-sharing technology.

Title: The Matrix
Genre: Sci-Fi
Year: 1999
Description: A hacker discovers reality is a simulated
world.

Title: Avatar
Genre: Sci-Fi
Year: 2009
Description: A marine on an alien planet finds himself torn
between two worlds.
```

Question -C

Use bitwise operators to check if a given number is a power of 2. Write a function that returns True if the number is a power of 2, otherwise False.

```
def is_power_of_two(n):
    """Check if a number is a power of 2 using bitwise
operators."""
    return n > 0 and (n & (n - 1)) == 0

# Example usage:
numbers = [1, 2, 3, 4, 8, 12, 16, 31]
for num in numbers:
    print(f"{num} is a power of 2: {is_power_of_two(num
)}")
```

Output:

```
1 is a power of 2: True
2 is a power of 2: True
3 is a power of 2: False
4 is a power of 2: True
8 is a power of 2: True
12 is a power of 2: False
16 is a power of 2: True
31 is a power of 2: False
```