```
# Import 'math' to do the mathematical calculations
import math
# The JSON for data parsing
import json

# This is a function that calculates the distance between two points on a sphere using the great-circle formula.
def calculate_distance(lat1, lon1, lat2, lon2):
    # Convert degrees to radians
    lat1, lon1, lat2, lon2 = map(math.radians, [lat1, lon1, lat2, lon2])

    # Radius of the Earth in kilometers
    radius = 6371

    # The Haversine formula is a mathematical formula used to calculate the distance between two points on a sphere, such as the Eart
    dlat = lat2 - lat1
    dlon = lon2 - lon1
    a = math.sin(dlat/2)**2 + math.cos(lat1) * math.cos(lat2) * math.sin(dlon/2)**2
    c = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))
    distance = radius * c

    return distance

# St Stephen's Green park coordinates
park_lat = 53.337839
park_lon = -6.259520

# Read and process friends' records from the file
matching_friends = []

with open("friends (1).txt", "r") as file:
    for line in file:
        try:
            friend_data = json.loads(line)  # Parse JSON-like data
            friend_lat = float(friend_data["latitude"])
            friend_lon = float(friend_data["longitude"])

            # Find the distance from the friend to St Stephen's Green park
```

```
            # Find the distance from the friend to St Stephen's Green park
            distance = calculate_distance(friend_lat, friend_lon, park_lat, park_lon)

            # Verify if the friend is located within a distance of 100 kilometers and include them in the list.
            if distance <= 100:
                matching_friends.append((friend_data["name"], friend_data["user_id"]))
        except (json.JSONDecodeError, KeyError):
            pass  # Skip invalid JSON data and data without latitude/longitude

# Arrange the friends who have a match in ascending order based on their User IDs
matching_friends.sort(key=lambda x: x[1])

# Display the names and user IDs of friends that are a match
for friend in matching_friends:
    print(f"Name: {friend[0]}, User ID: {friend[1]}")
```

```
Name: Ian Kehoe, User ID: 4
Name: Nora Dempsey, User ID: 5
Name: Theresa Leaon, User ID: 6
Name: Eoin Ahearn, User ID: 8
Name: Richard Finnegan, User ID: 11
Name: Christina MacFarlane, User ID: 12
Name: Olive Ahearn, User ID: 13
Name: Michael Ahearn, User ID: 15
Name: Patricia Cahill, User ID: 17
Name: Eoin Rosan, User ID: 23
Name: Rose Leaon, User ID: 24
Name: Stephen MacFarlane, User ID: 26
Name: Oliver Ahearn, User ID: 29
Name: Nick Leaon, User ID: 30
Name: Alan Behan, User ID: 31
Name: Lisa Ahearn, User ID: 39
```