TEST1: DATA STRUCTURES AND ALGORITHMS

INSTRUCTIONS:

1. **General Instructions:**
    - **Ensure you attempt all questions.**
    - **Ensure you follow the naming conventions as specified.**
2. **Submission of Practical Questions:**
    - **Your solutions for practical questions should be uploaded to your GitHub repositories.**
    - **Create a main folder for your submission named as: DSA-TEST1-[Your REGNO or ACCESS NUMBER].**
    - **Inside the main folder:**
        - **Name each file based on the number of the question you attempted (e.g., Question1.py, Question2.py).**
        - **Include a README.md file.**
3. **README File:**
    - **The README.md should provide an explanation of your work.**
    - **Clearly outline the steps to run each solution.**
    - **Any assumptions or specific libraries used should be mentioned.**
4. **Final Steps:**
    - **Once you've added all your solutions and the README file, push your folder to your GitHub repository.**
    - **Ensure that your repository is public so that it can be accessed and evaluated.**

Remember to review your work before submission. Best of luck!

Question 1.

```python
lst = [14, 8, -23, 4, 6, 10, -18, 5, 5, 11]
maxSum = lst[0]  # Initialize maxSum with the first element of the list
sumz = 0

for i in range(0, len(lst)):
    sumz += lst[i]

    if sumz < 0:
        sumz = 0
    elif maxSum < sumz:
        maxSum = sumz

print("Maximum Sub Array Sum =", maxSum)

```

a) What type of data structure is being used to store the numbers in the code, and why is this data structure suitable for the task?

b) The code initializes maxSum with the first element of the list. Can you explain the significance of this initialization and how it affects the algorithm's correctness?

c) How does the for loop iterate through the list, and what is the purpose of the sumz variable? How is it being updated within the loop?

d) Explain how the code identifies the maximum subarray sum. What conditions trigger the update of the maxSum variable?

e) What is the time complexity of this code for finding the maximum subarray sum, and how does the choice of data structure and algorithm contribute to its efficiency or performance?

2. **Question:**

The Uganda Martyrs are a significant part of the history of Uganda, consisting of 23 Anglican and 22 Catholic converts who were executed between 31 January 1885 and 27 January 1887 in the historical kingdom of Buganda.

Given the following list of names:

Catholic Martyrs:

Achileo Kiwanuka
Adolphus Ludigo-Mukasa
Ambrosius Kibuuka
Anatoli Kiriggwajjo
Andrew Kaggwa
Antanansio Bazzekuketta
Bruno Sserunkuuma
Charles Lwanga
Denis Ssebuggwawo
Wasswa
Gonzaga Gonza
Gyavira Musoke
Yowana Mukiibi
Yusufu Lugalama
Zakayo Lubega
James Buuzaabalyaawo
John Maria Muzeeyi
Joseph Mukasa Kizito
Lukka Baanabakintu
Matiya Mulumba
Mbaga Tuzinde
Mugagga Lubowa
Mukasa Kiriwawanvu
Nowa Mawaggali
Ponsiano Ngondwe

Anglican Martyrs:

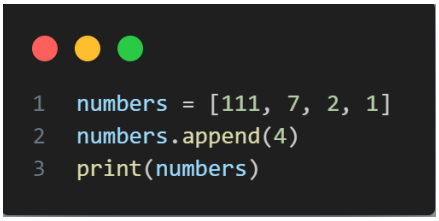
Aaron Lubega
Apollo Kivebulaya
Eria Sebukyala
Fredrick Kizza
George Kizza
James Hannington
Janani Luwum
Joseph Balikuddembe
Kizito
Mark Kakumba
Matia Mulumba
Nuhu Mbegu
Robert Munyagayirwa
Samwiri Mukasa
Yefusa Namayanja
Yohana Mukasa
Yosefu Lugalama
Yowana Kitaka
Yowana Maria Mukasa

Tasks:

1. Create two separate lists named **catholic_martyrs** and **anglican_martyrs** containing the names of Catholic and Anglican martyrs respectively.
2. Identify and remove any duplicate names present in both lists.
3. Write a function named **martyr_count** that takes in a martyr's name as an argument and returns the group (Catholic or Anglican) to which the martyr belongs.
4. Using the function from task 3, determine the group of the martyr named "Kizito".
5. Write a Python function that returns True if the input string matches names of the Uganda Martyrs in both lists

OBJECTIVES

1. **Question**: Which method is used to add an element to the end of a Python list?
    - a) **insert()**
    - b) **add()**
    - c) **append()**
    - d) **put()**
2. **Question**: How do you add an element 'x' at the beginning of a list named **my_list**?
    - a) **my_list.add(0, 'x')**
    - b) **my_list.insert(0, 'x')**
    - c) **my_list.append('x', 0)**
    - d) **my_list.append(0, 'x')**
3. **Question**: What does the following code output?



```
numbers = [111, 7, 2, 1]
numbers.append(4)
print(numbers)
```

    - a) **[111, 7, 2, 1]**
    - b) **[111, 7, 2, 1, 4]**
    - c) **[4, 111, 7, 2, 1]**
    - d) **Error**
4. **Question**: Which method is used to copy the entire content of a list **my_list**?
    - a) **my_list.copy()**
    - b) **my_list[:]**
    - c) **list(my_list)**
    - d) All of the above
5. **Question**: How can you remove the first element from a Python list named **my_list**?
    - a) **my_list.remove(0)**
    - b) **del my_list[1]**
    - c) **my_list.pop(0)**

- d) **my_list.delete(0)**

6. **Question**: If you have a list **l1**, what does **l2 = l1** do?
   - a) Creates a new list **l2** with the same content as **l1**
   - b) Makes **l1** and **l2** point to two different lists
   - c) Makes **l1** and **l2** point to the same list
   - d) Gives an error
7. **Question**: Which of the following will check if 'apple' is NOT in the list **fruits**?
   - a) **'apple' not in fruits**
   - b) **fruits not contains 'apple'**
   - c) **'apple' in not fruits**
   - d) **fruits contains not 'apple'**
8. **Question**: What is the output of the following code?

```
my_list = [1, 2, 3, 4, 5]
del my_list[0:2]
print(my_list)
```

   - a) **[1, 2, 3, 4, 5]**
   - b) **[3, 4, 5]**
   - c) **[1, 2]**
   - d) **[4, 5]**
9. **Question**: How do you retrieve the last two elements of a list **my_list** using slicing?
   - a) **my_list[-2:]**
   - b) **my_list[-2, -1]**
   - c) **my_list[-1, -2]**
   - d) **my_list[-2, :]**
10. **Question**: If **my_list = ["A", "B", 1, 2]**, what is the output of **print(2 in my_list)**?
    - a) **True**
    - b) **False**
    - c) **1**
    - d) **Error**