

Python Review

1D List, 2D LIST

Outline

➤ 1D LIST

➤ 2D LIST

1D LIST – Basic Exercise

Bài 1: Bạn hãy tạo 1D List gồm 5 phần tử [1, 2, 3, 4, 5]. Sau đó in ra màn hình 1D list vừa tạo ?

□ Cách 1:

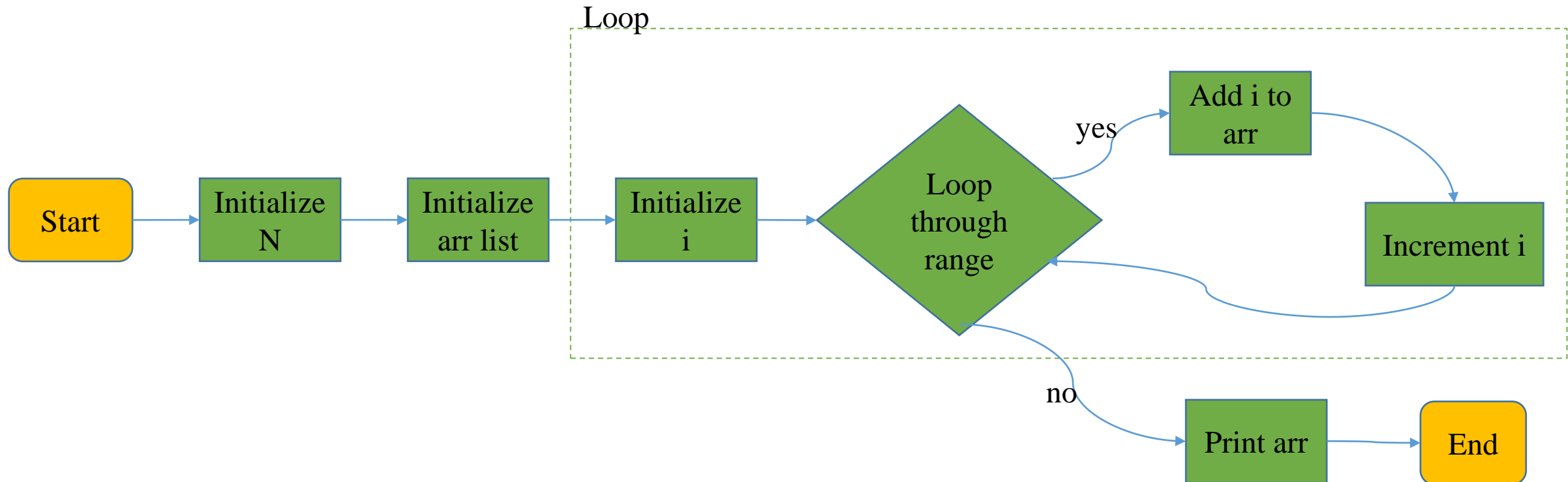
 Ý tưởng:



1D LIST – Basic Exercise

❑ Cách 2: Sử dụng comprehension

💡 Ý tưởng:



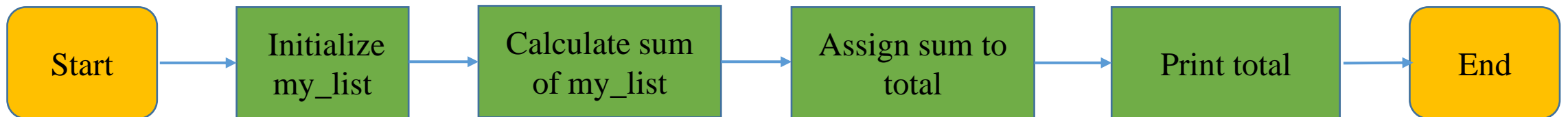
1D LIST – Basic Exercise

Bài 2: Bạn hãy xây dựng chương trình thực hiện tính tổng các phần tử trong 1D List?

- Input: `my_list = [1, 2, 3, 4, 5]`
- Output: 15



Ý tưởng:



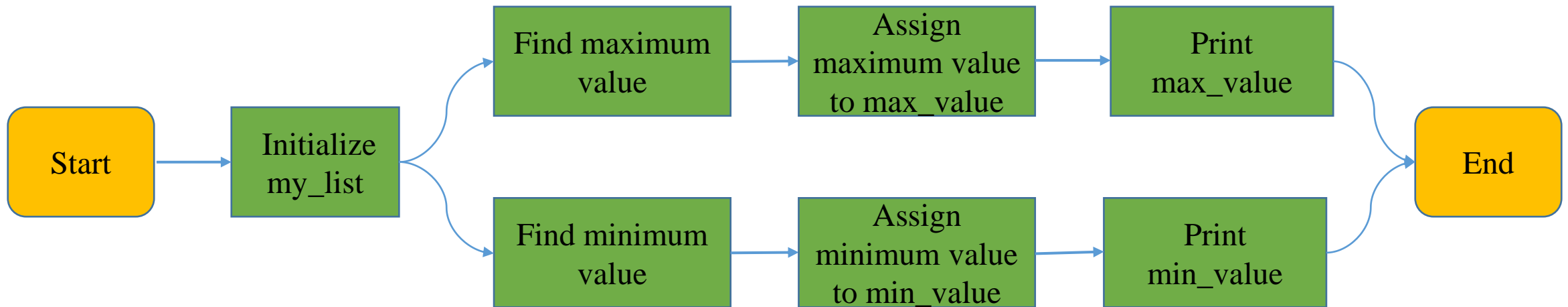
1D LIST – Basic Exercise

Bài 3: Bạn hãy xây dựng chương trình tìm giá trị lớn nhất và nhỏ nhất trong danh sách 1d?

- Input: `my_list = [1, 2, 3, 4, 5]`
- Output: `max = 5, min = 1`



Ý tưởng:

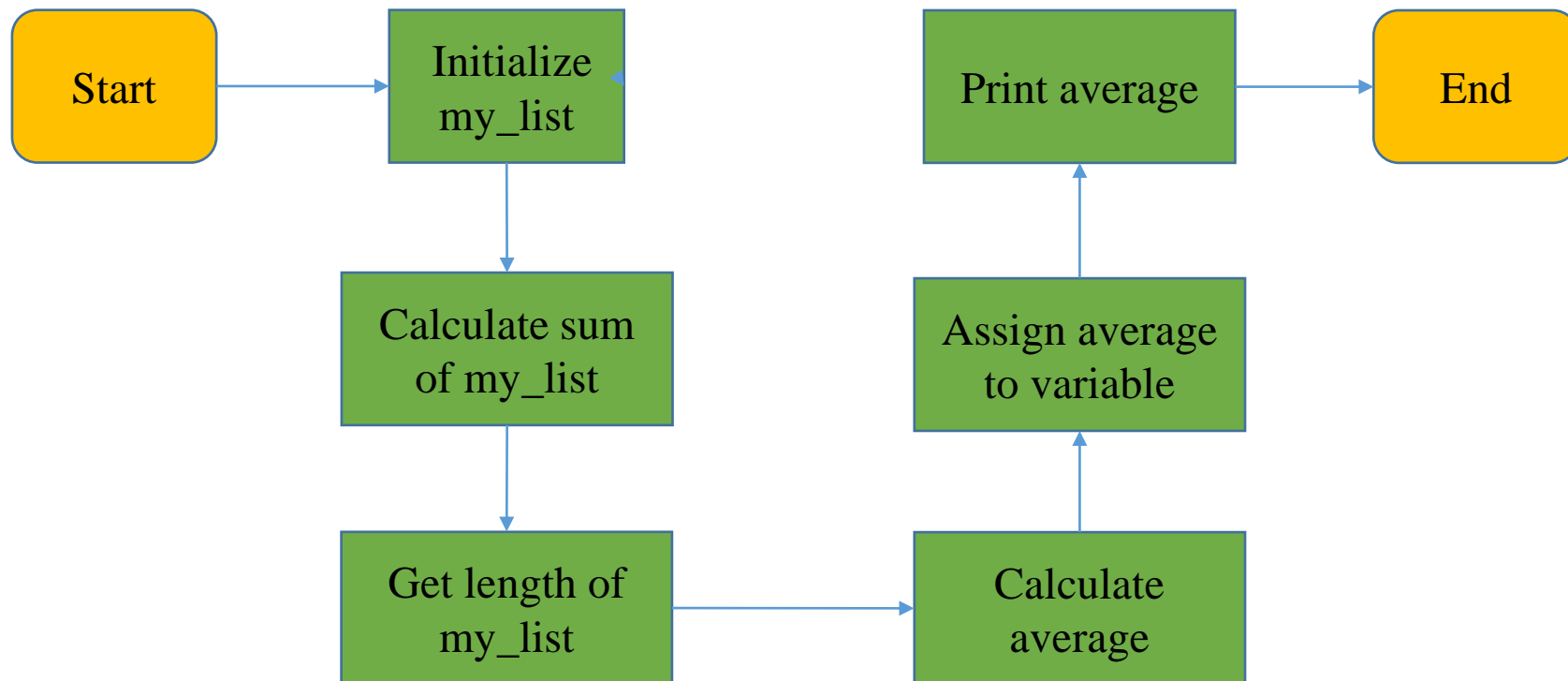


1D LIST – Basic Exercise

Bài 4: Bạn hãy xây dựng chương trình tính trung bình cộng của 1D list?

- Input: `my_list = [1, 2, 3, 4, 5]`
- Output: 3.0

 Ý tưởng:



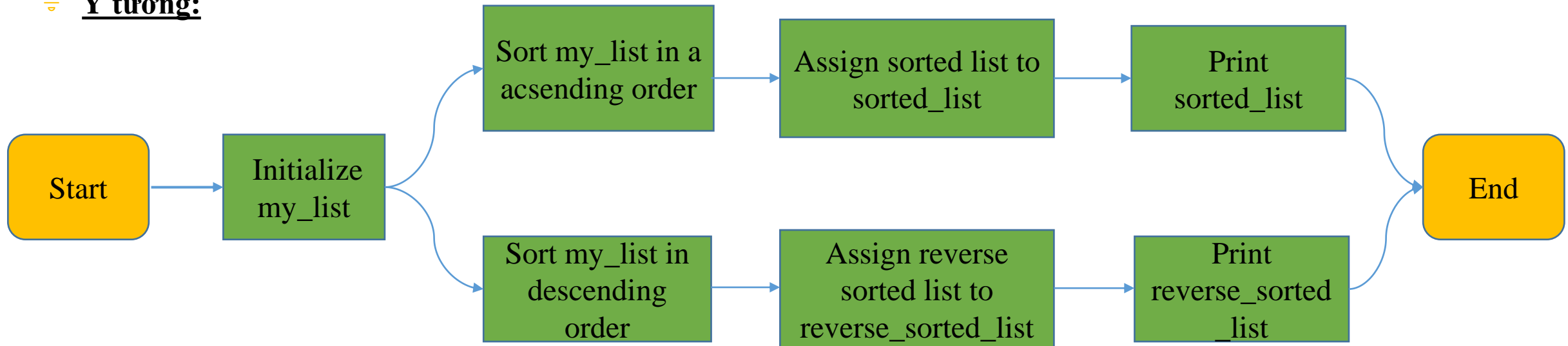
1D LIST – Basic Exercise

Bài 5: Bạn hãy xây dựng chương trình sắp xếp 1D List sau theo chiều tăng dần hoặc giảm dần?

- Input: `my_list = [5, 3, 1, 4, 2]`
- Output: tăng dần `[1, 2, 3, 4, 5]`, giảm dần: `[5, 4, 3, 2, 1]`



Ý tưởng:



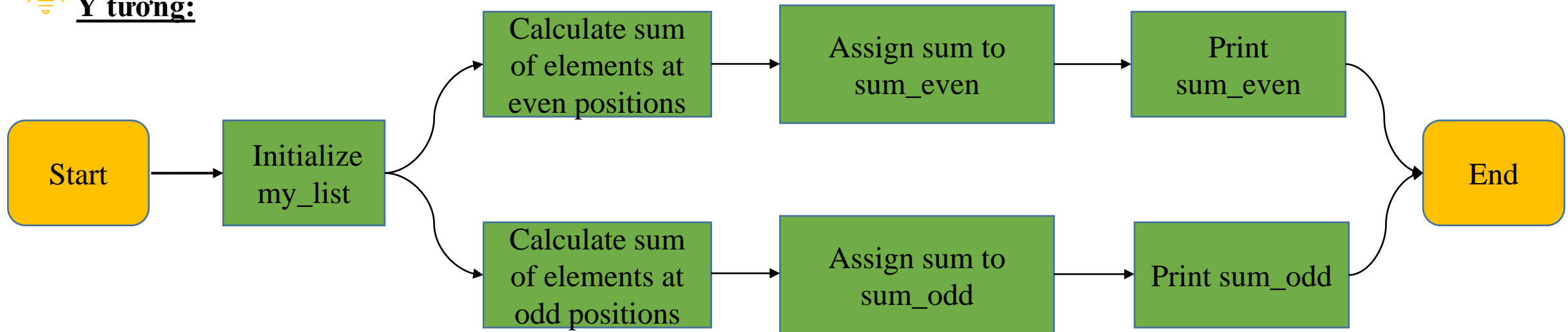
1D LIST – Basic Exercise

Bài 6: Bạn hãy xây dựng chương trình tổng các phần tử trên các vị trí chẵn hoặc lẻ?

- Input: `my_list = [1, 2, 3, 4, 5]`
- Output: `even = 6, odd = 9`



Ý tưởng:



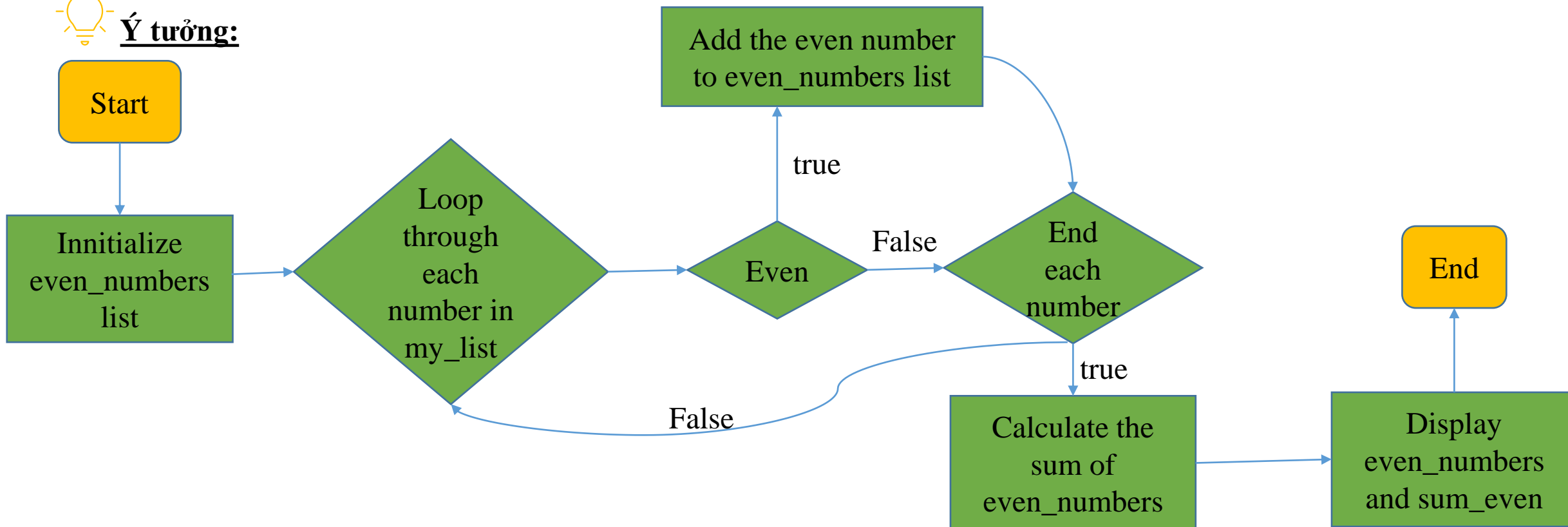
1D LIST - Advanced

Bài 1: Bạn hãy xây dựng chương trình tìm các số chẵn trong danh sách và tính tổng của chúng ?

- Input: `my_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]`
- Output: `even_numbers = [2, 4, 6, 8, 10]`, `sum_even = 30`



Ý tưởng:



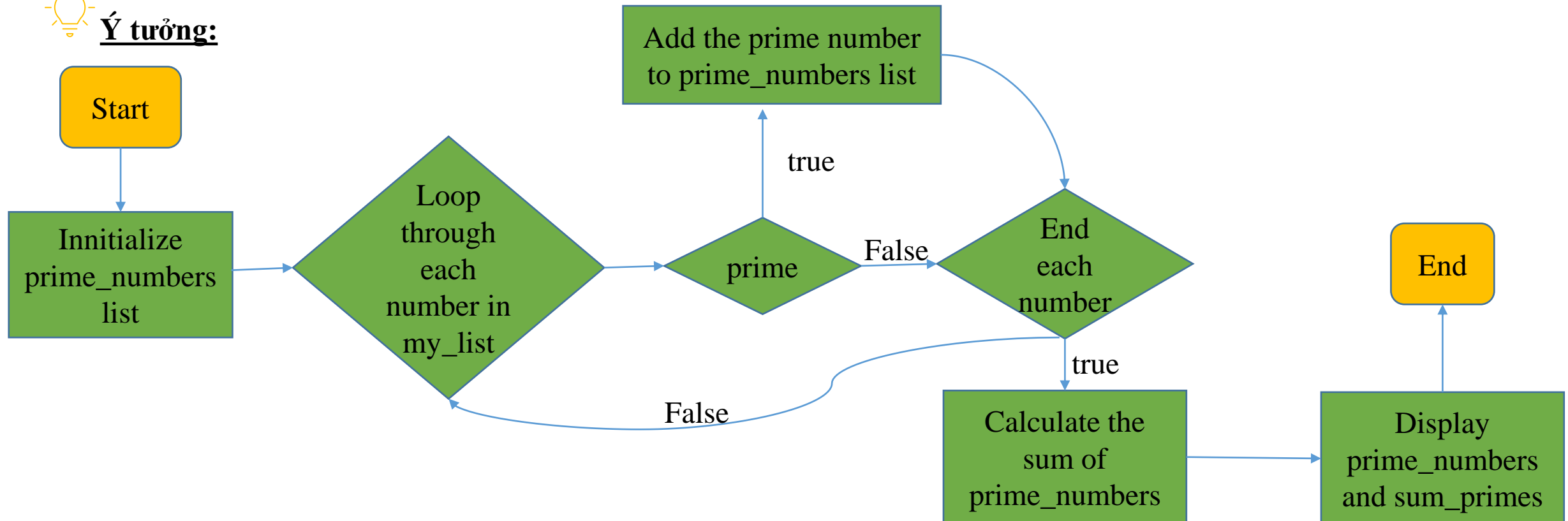
1D LIST - Advanced

Bài 2: Bạn hãy xây dựng chương trình tính tổng của các số nguyên tố trong danh sách?

- Input: `my_list = [2, 3, 4, 5, 6, 7, 8, 9, 10]`
- Output: `prime_numbers = [2, 3, 5, 7]`, `sum_prime = 17`



Ý tưởng:



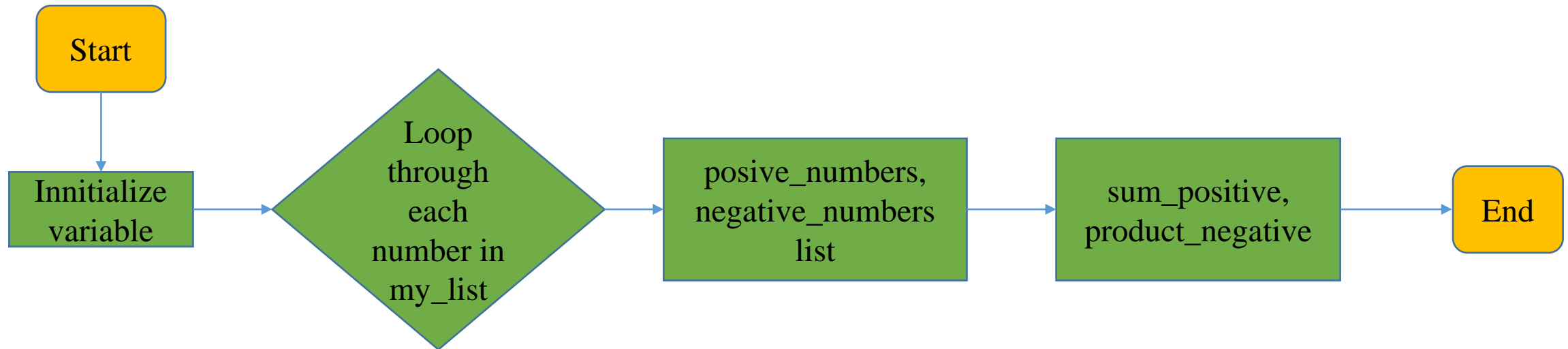
1D LIST - Advanced

Bài 3: Bạn hãy xây dựng chương trình tính tổng các số dương và tích các số âm trong danh sách:?

- Input: `my_list = [-1, 2, -3, 4, -5, 6]`
- Output: `sum_positive = [12]`, `product_negative = -15`



Ý tưởng:



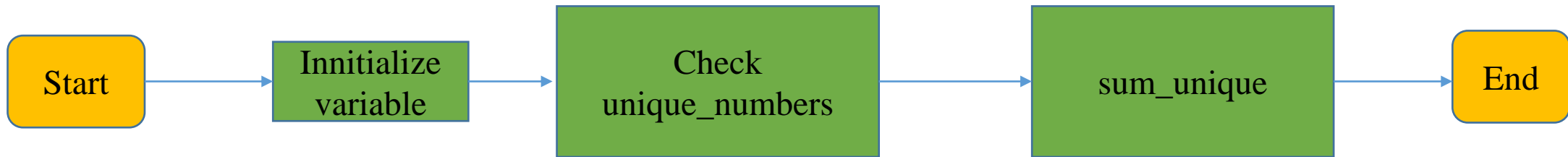
1D LIST - Advanced

Bài 4: Bạn hãy xây dựng chương trình tính tổng các số trong danh sách sau khi loại bỏ các số trùng lặp?

- Input: `my_list = [1, 2, 3, 2, 4, 5, 1, 6, 7, 8, 5]`
- Output: `unique_numbers = [1, 2, 3, 4, 5, 6, 7, 8]`, `sum_unique = 36`



Ý tưởng:



Outline

➤ 1D LIST

➤ 2D LIST

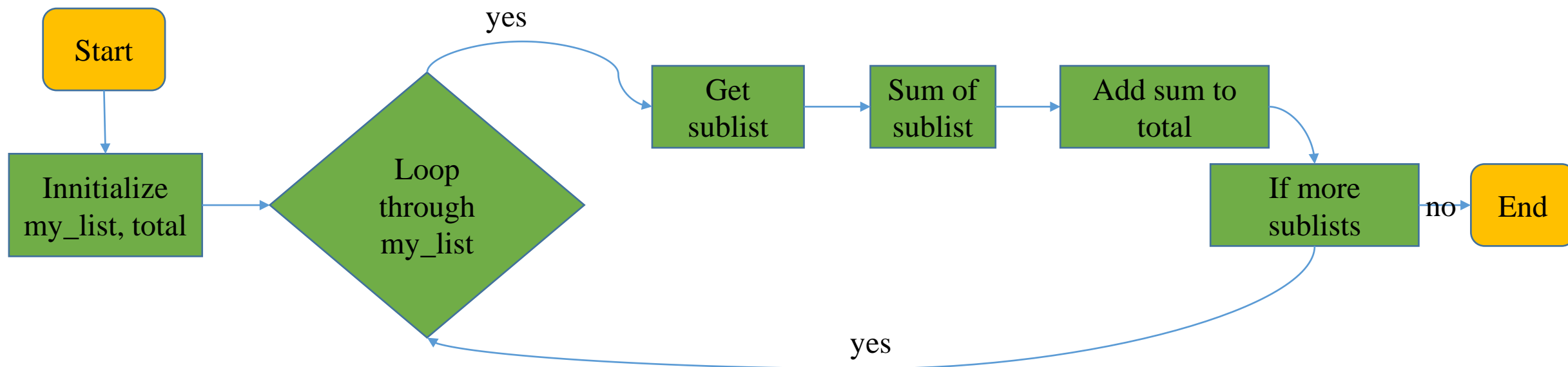
2D LIST - Basic

Bài 1: Bạn hãy xây dựng chương trình tính tổng các phần tử trong danh sách 2D?

- Input: `my_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]`
- Output: `total = 45`



Ý tưởng:



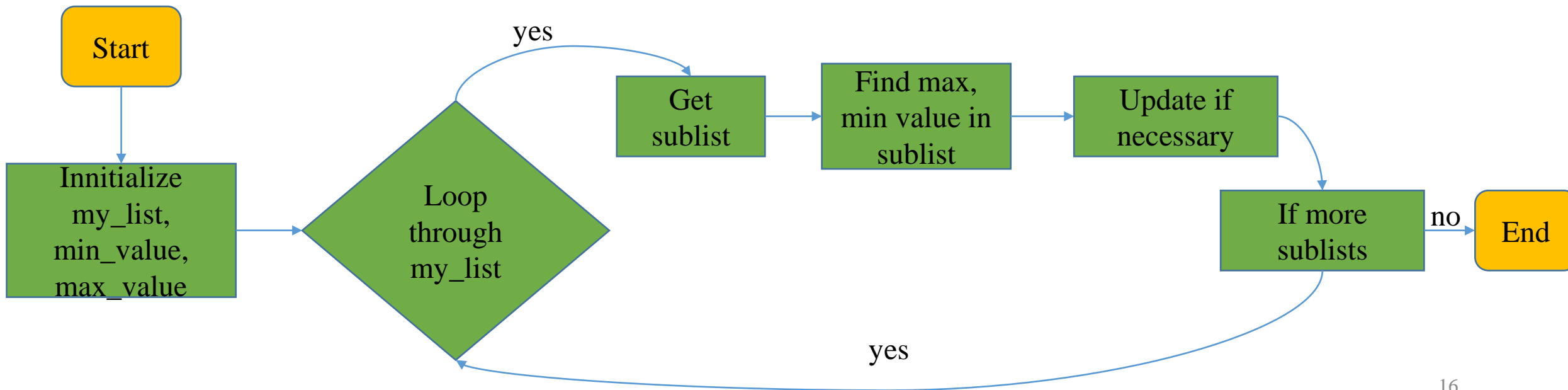
2D LIST - Basic

Bài 2: Bạn hãy xây dựng chương trình tìm giá trị lớn nhất và nhỏ nhất trong danh sách 2D?

- Input: `my_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]`
- Output: `max_value = 9, min_value = 1`



Ý tưởng:



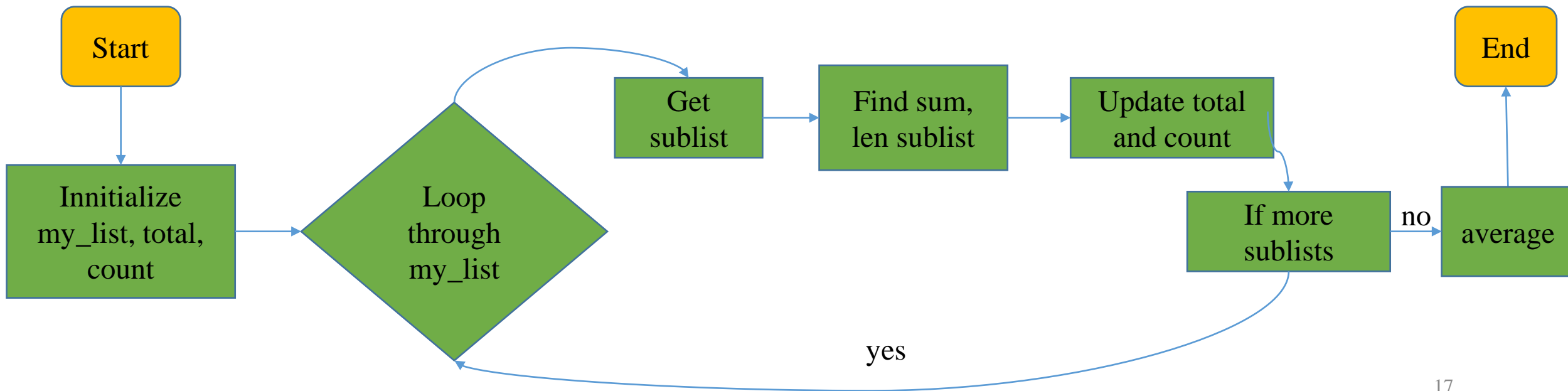
2D LIST - Basic

Bài 3: Bạn hãy xây dựng chương trình tính trung bình cộng của danh sách 2D?

- Input: `my_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]`
- Output: `average = 5.0`



Ý tưởng:



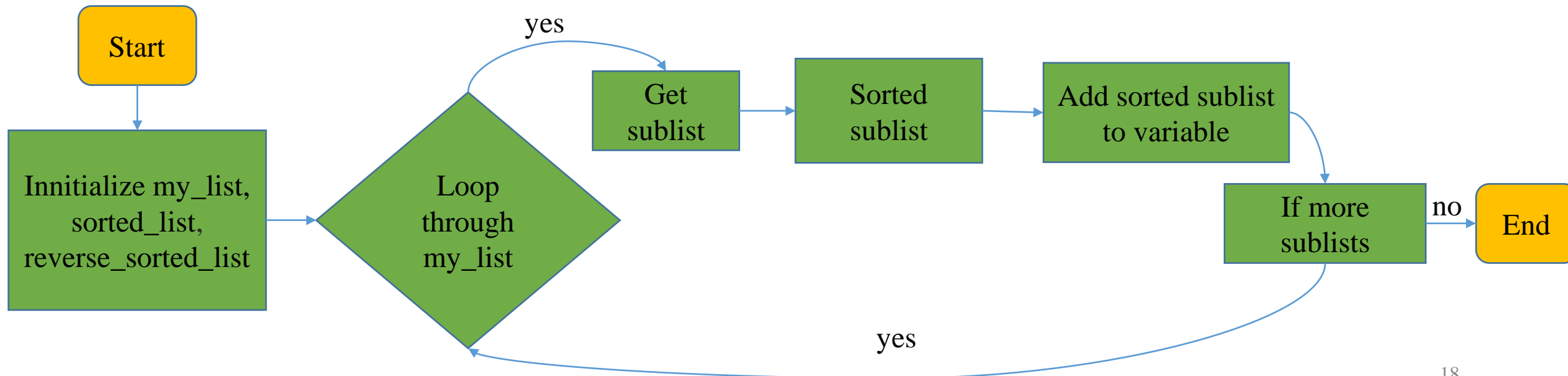
2D LIST - Basic

Bài 4: Bạn hãy xây dựng chương trình sắp xếp danh sách 2D theo thứ tự tăng dần hoặc giảm dần?

- Input: `my_list = [[5, 3, 1], [4, 2, 6], [9, 7, 8]]`
- Output: `sorted_list = [[1, 3, 5], [2, 4, 6], [7, 8, 9]]`,
`reverse_sorted_list = [[5, 3, 1], [6, 4, 2], [9, 8, 7]]`



Ý tưởng:

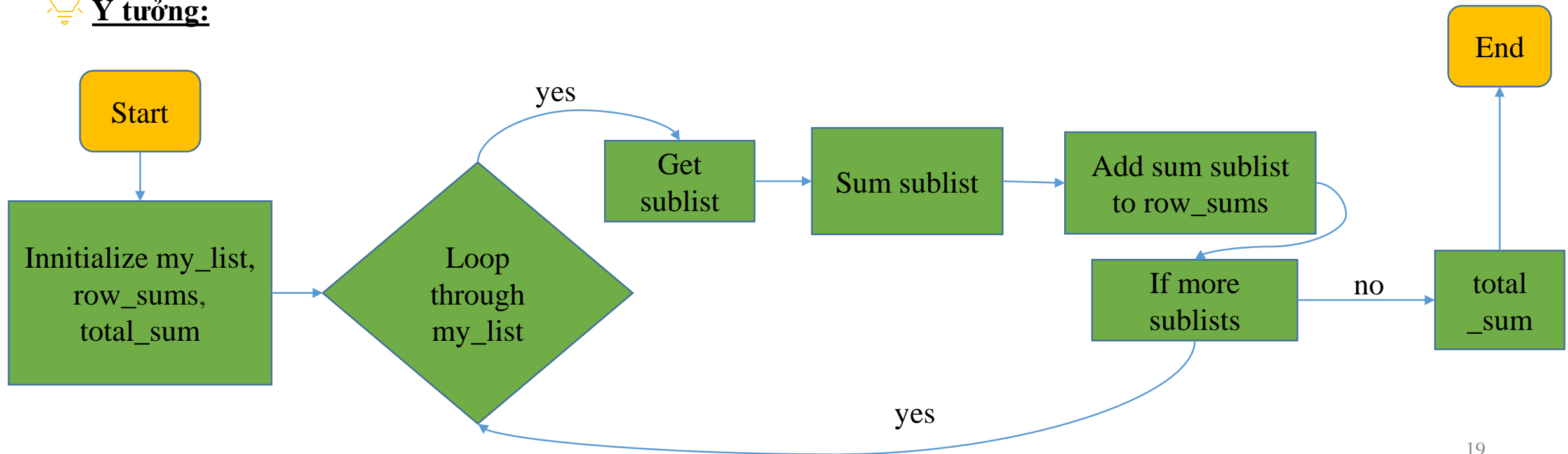


2D LIST - Advanced

Bài 1: Bạn hãy xây dựng chương trình tính tổng các phần tử trong mỗi hàng và tính tổng của các tổng hàng?

- Input: `my_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]`
- Output: `row_sums = [6, 15, 24]`, `total_sum = 45`

💡 Ý tưởng:



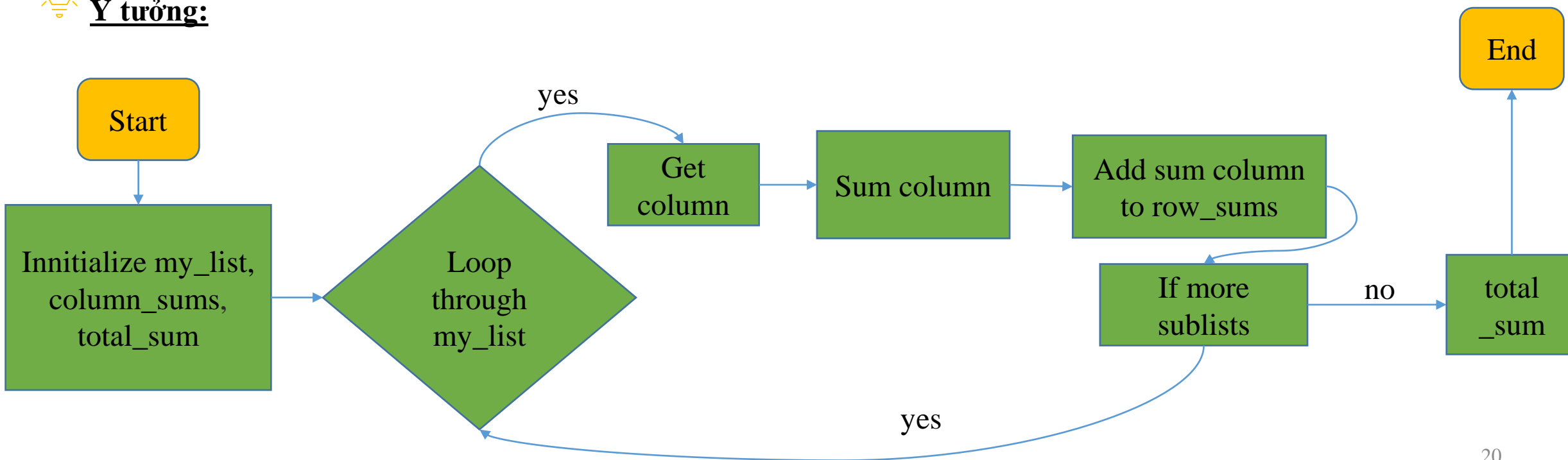
2D LIST - Advanced

Bài 2: Bạn hãy xây dựng chương trình tính tổng các phần tử trong mỗi cột và tính tổng của các tổng cột?

- Input: `my_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]`
- Output: `column_sums = [12, 15, 18]`, `total_sum = 45`



Ý tưởng:



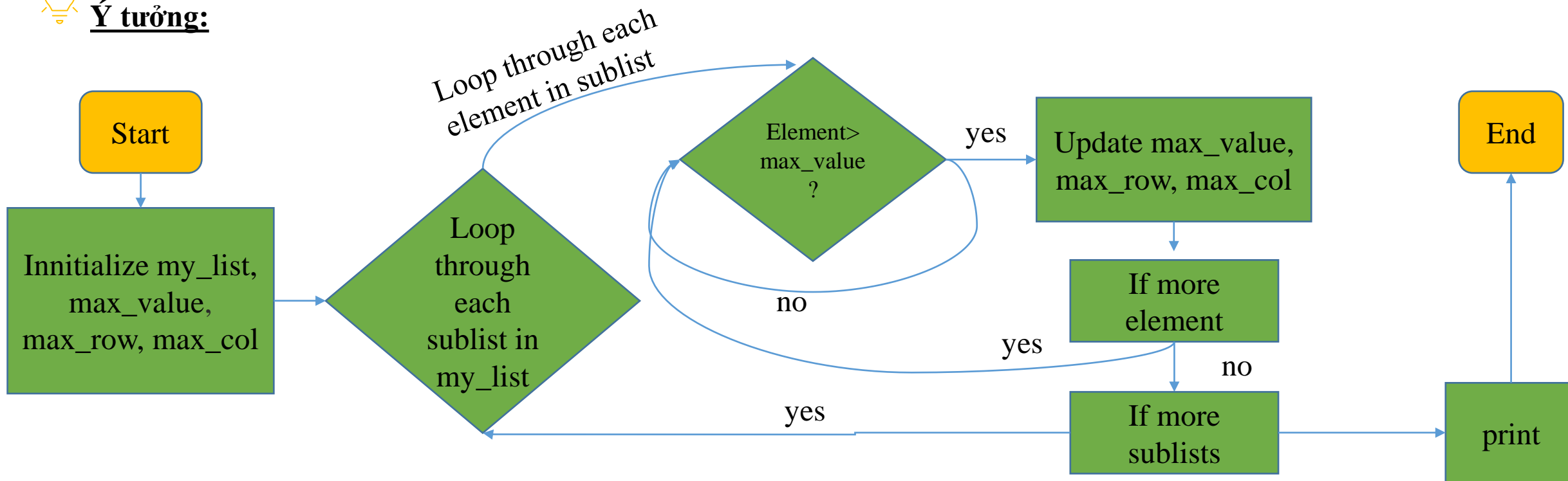
2D LIST - Advanced

Bài 3: Bạn hãy xây dựng chương trình tìm phần tử lớn nhất và vị trí của phần tử đó trong danh sách 2D?

- Input: `my_list = [[1, 2, 3], [4, 9, 6], [7, 8, 5]]`
- Output: `max_value = 9; max_row, max_col = 1, 1`



Ý tưởng:



2D LIST - Advanced

Bài 4: Bạn hãy xây dựng chương trình tìm phần tử lớn nhất trong danh sách 2D và hiển thị hàng chứa phần tử đó?

- Input: `my_list = [[1, 2, 3], [4, 9, 6], [7, 8, 5]]`
- Output: `max_value = 9, max_row = [4, 9, 6]`



Ý tưởng:

