

Data Structure & Algorithm

Algorithm

Sorting

1. Bubble Sort: Sắp xếp nổi bọt

➤ Nguyên lý:

- ✓ Chạy từ đầu đến cuối mảng.
- ✓ Nếu phần tử đứng trước mà lớn hơn phần tử đứng sau thì đổi chỗ.
- ✓ Sau mỗi vòng lặp thì phần tử lớn nhất sẽ trôi xuống dưới.

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 1



i $i+1$



i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 1



i $i+1$



i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 1

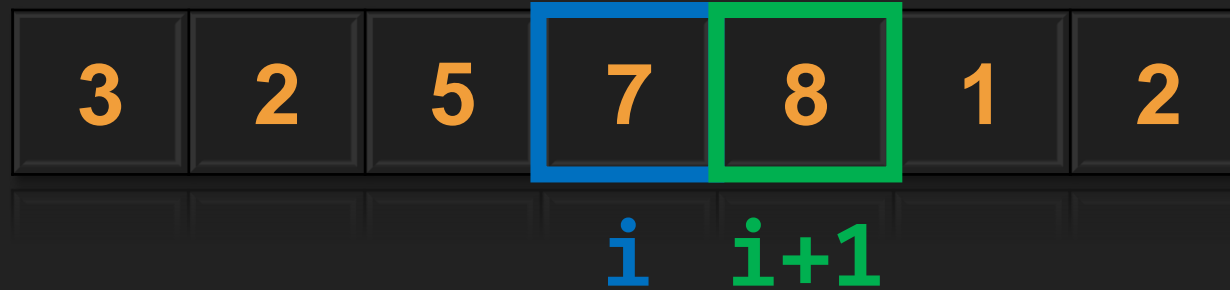


1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 1



1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 1



i $i+1$

i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 1



i $i+1$



i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 2



i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 2

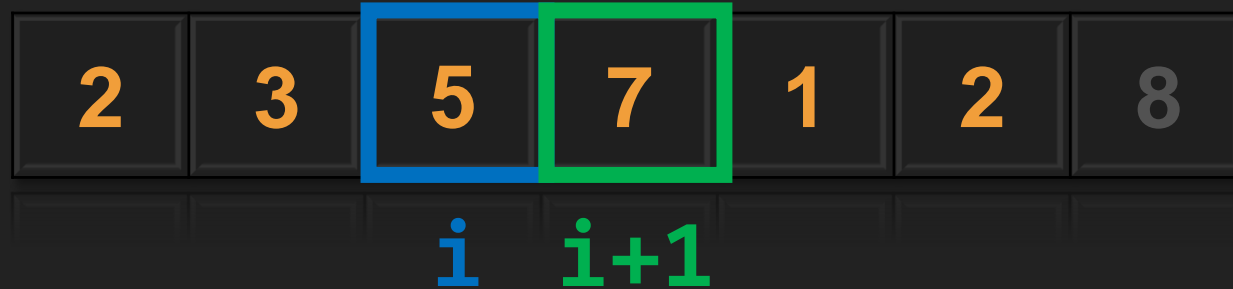


1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 2



1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 2



i $i+1$



i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 2



i $i+1$



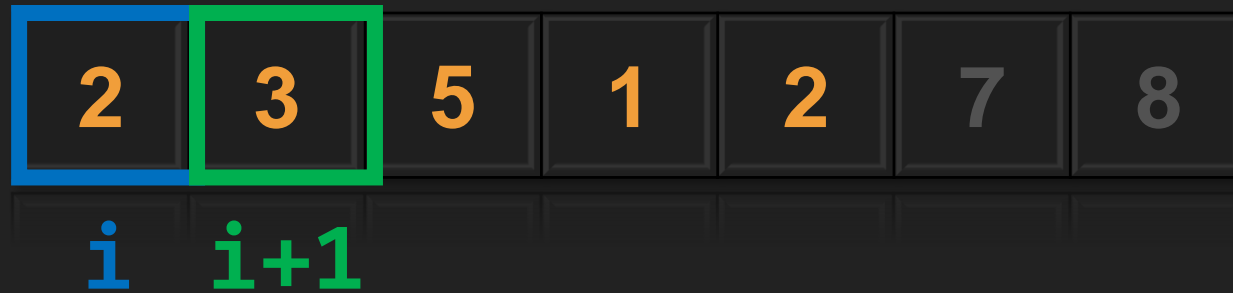
i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 3

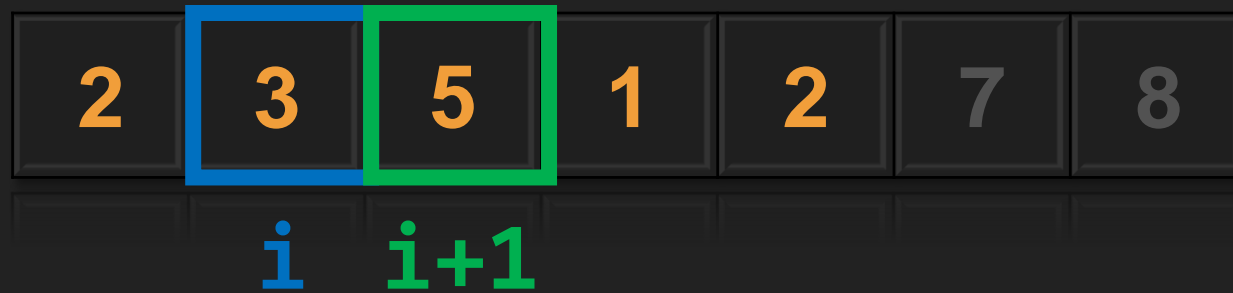


1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 3



1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 3



i $i+1$



i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

➤ Mô phỏng:



Loop 3



i $i+1$



i $i+1$

1. Bubble Sort: Sắp xếp nổi bọt

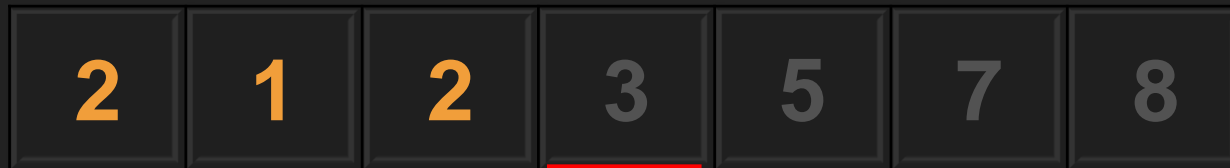
➤ Mô phỏng:



Loop 3



Loop 4

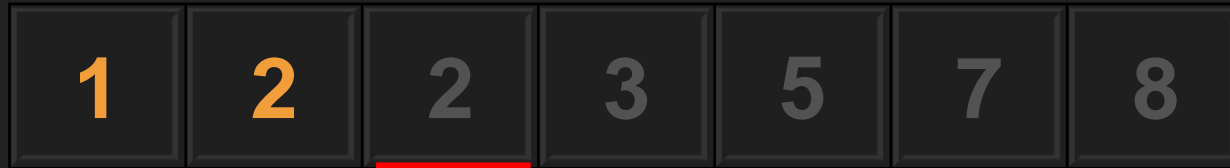


1. Bubble Sort: Sắp xếp nổi bọt

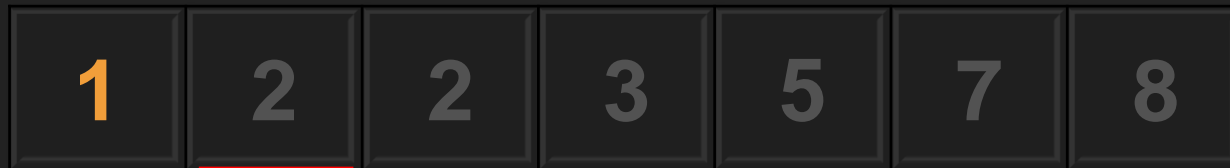
➤ Mô phỏng:



Loop 5



Loop 6



1. Bubble Sort: Sắp xếp nổi bọt

➤ Code:

1. Bubble Sort: Sắp xếp nổi bọt

➤ Nâng cấp:

- ✓ Dừng lại khi dãy đã được sắp xếp xong.

2. Insertion Sort: Sắp xếp chèn

➤ Nguyên lý:

- ✓ Chạy từ đầu đến cuối mảng.
- ✓ Tại vòng lặp i , coi như dãy từ $[0, i-1]$ đã được sắp xếp, chèn phần tử $a[i]$ vào vị trí thích hợp.
- ✓ Sau vòng lặp thứ i thì dãy $[0, i]$ đã được sắp xếp.

2. Insertion Sort: Sắp xếp chèn

➤ Mô phỏng:



Loop 1



i

2. Insertion Sort: Sắp xếp chèn

➤ Mô phỏng:



Loop 2



i



i

2. Insertion Sort: Sắp xếp chèn

➤ Mô phỏng:



Loop 3



i

2. Insertion Sort: Sắp xếp chèn

➤ Mô phỏng:



Loop 4



i

2. Insertion Sort: Sắp xếp chèn

➤ Mô phỏng:



Loop 5



i

i

2. Insertion Sort: Sắp xếp chèn

➤ Mô phỏng:



Loop 5



i



i

2. Insertion Sort: Sắp xếp chèn

➤ Code:

3. Selection Sort: Sắp xếp chọn

➤ Nguyên lý:

- ✓ Chạy từ đầu đến cuối mảng.
- ✓ Tại vòng lặp thứ i , tìm phần tử nhỏ nhất trong khoảng $[i+1, n-1]$, nếu nhỏ hơn $a[i]$ thì đổi chỗ cho $a[i]$.
- ✓ Sau vòng lặp thứ i , thì dãy $[0, i]$ đã được sắp xếp.

3. Selection Sort: Sắp xếp chọn

➤ Mô phỏng:

5	3	2	7	8	1	2
---	---	---	---	---	---	---

Loop 1

5	3	2	7	8	1	2
---	---	---	---	---	---	---

i

j



1	3	2	7	8	5	2
---	---	---	---	---	---	---

i

j

3. Selection Sort: Sắp xếp chọn

➤ Mô phỏng:



Loop 2



i

j



i

j

3. Selection Sort: Sắp xếp chọn

➤ Mô phỏng:

5	3	2	7	8	1	2
---	---	---	---	---	---	---

Loop 3

1	2	3	7	8	5	2
---	---	---	---	---	---	---

i

j



1	2	2	7	8	5	3
---	---	---	---	---	---	---

i

j

3. Selection Sort: Sắp xếp chọn

➤ Mô phỏng:



Loop 4



j



i

j

3. Selection Sort: Sắp xếp chọn

➤ Mô phỏng:



Loop 5



i

j

i

j

3. Selection Sort: Sắp xếp chọn

➤ Mô phỏng:

5	3	2	7	8	1	2
---	---	---	---	---	---	---

Loop 6

1	2	2	3	5	8	7
---	---	---	---	---	---	---

i

j



1	2	2	3	5	7	8
---	---	---	---	---	---	---

i

j

3. Selection Sort: Sắp xếp chọn

➤ Code:

4. Bài tập

✓ 414. Third Maximum Number

✓ 1051. Height Checker

✓ 1636. Sort Array by Increasing Frequency

3. Chuẩn bị buổi sau:

- '- Kim Tùng: Dãy số Fibonacy là gì? Tính số Fibonacy bằng mảng (ko đệ quy) và bằng đệ quy
- Hữu Tín: Cách tính giai thừa ($n!$) và tính tổng ($a[]$) = đệ quy

Data Structure & Algorithm

