

## BÀI 3

# TÌM KIẾM – SẮP XẾP

Email: [anvanminh.hau@gmail.com](mailto:anvanminh.hau@gmail.com)

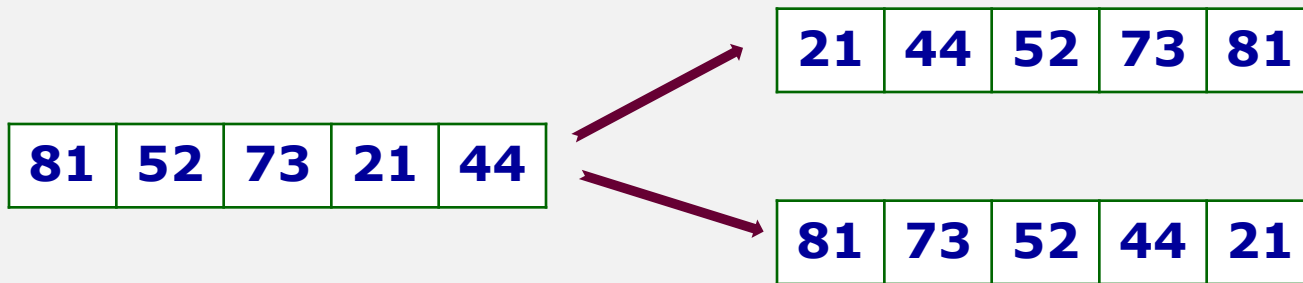
# Nội dung

1. Khái niệm sắp xếp
2. Phát biểu bài toán sắp xếp
3. Sắp xếp trong và sắp xếp ngoài
4. Các thuật toán sắp xếp đơn giản
  - Sắp xếp nổi bọt – bubble sort
  - Sắp xếp lựa chọn – selection sort
  - Sắp xếp chèn – insertion sort
5. Sắp xếp nâng cao
  - Sắp xếp phân đoạn – quick sort
  - Sắp xếp vun đống – heap sort
  - Sắp xếp trộn – merge sort

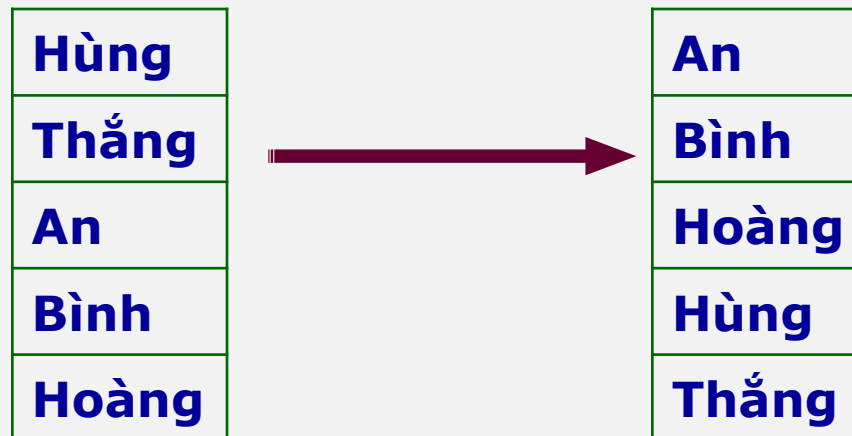
# 1. Khái niệm sắp xếp

- Vấn đề sắp xếp

- Sắp xếp dãy số



- Sắp xếp danh sách học sinh



# Khái niệm sắp xếp (tt)

- **Khái niệm**

- Sắp xếp là việc biến đổi vị trí của một tập đối tượng theo một trật tự mới nhằm thỏa mãn một mục đích.

- **Mục đích**

- Giúp việc tìm kiếm được thực hiện nhanh hơn.
- Sắp thứ tự để in bảng biểu.

## 2. Bài toán sắp xếp



### Đầu vào:

Dãy  $n$  đối tượng, mỗi đối tượng có một khóa sắp xếp.

### Đầu ra:

Dãy  $n$  đối tượng được sắp xếp theo trật tự của khóa.

## 2. Bài toán sắp xếp



**Đầu vào:**

x[0]	x[1]	x[2]	x[3]	x[4]
5	-1	7	3	-4

**Đầu ra:**

x[0]	x[1]	x[2]	x[3]	x[4]
-4	-1	3	5	7

### 3. Khái niệm sắp xếp trong và sắp xếp ngoài

- **Sắp xếp trong:**

- Dãy đối tượng được sắp có mặt đầy đủ ở bộ nhớ trong (RAM).
- Sắp xếp danh sách được lưu trong mảng một chiều: danh sách các số, danh sách học sinh, ...
- Sắp tại chỗ: Đổi vị trí các phần tử trong mảng.

- **Sắp xếp ngoài:**

- Dãy đối tượng được sắp chưa có mặt đầy đủ ở bộ nhớ trong.
- Sắp xếp dữ liệu được lưu trong tệp.

## 4. Ba phương pháp sắp xếp cơ bản

1. Sắp xếp nổi bọt – Bubble sort
2. Sắp xếp lựa chọn – Selection sort
3. Sắp xếp chèn – Insertion sort



## 4.1. Sắp xếp nổi bọt – Bubble sort

- Ý tưởng giải thuật

- Thực hiện việc đổi chỗ các cặp phần tử kế tiếp trong dãy đối tượng cần sắp nếu chúng chưa đúng thứ tự.

- Mô tả việc sắp xếp:

- Cho dãy số có 5 số nguyên ( $n = 5$ ) được lưu trong mảng  $x$  như dưới đây.

- Dãy  $x$ :

$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$
42	23	74	11	65	58	34

- Yêu cầu: Mô tả việc Sắp xếp dãy số theo chiều tăng dần.

# Sắp xếp nổi bọt (tt) – Mô tả

- Duyệt lần 1:** So sánh các cặp phần tử kế tiếp của dãy là:  $(x[6], x[5]); (x[5], x[4]); \dots; (x[1], x[0])$ , nếu cặp nào trái thứ tự thì đảo giá trị.

$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$
42	23	74	11	65	58	34
42	23	74	11	65	34	58
42	23	74	11	34	65	58
42	23	74	11	34	65	58
42	23	11	74	34	65	58
42	11	23	74	34	65	58
11	42	23	74	34	65	58

# Sắp xếp nổi bọt (tt) – Mô tả

- Sau lần duyệt thứ nhất ta thấy giá trị nhỏ nhất trong dãy được chuyển về vị trí đầu tiên, nghĩa là nó đứng đúng vị trí.
- Các phần tử còn lại vẫn chưa đúng vị trí.
- Duyệt lần 2

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
11	42	23	74	34	65	58
11	42	23	74	34	58	65
11	42	23	74	34	58	65
11	42	23	34	74	58	65
11	42	23	34	74	58	65
11	23	42	34	74	58	65

# Sắp xếp nổi bọt (tt) – Mô tả

- Duyệt lần 3

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
11	23	42	34	74	58	65
11	23	42	34	74	58	65
11	23	42	34	58	74	65
11	23	42	34	58	74	65
11	23	34	42	58	74	65

# Sắp xếp nổi bọt (tt) – Mô tả

- Duyệt lần 4

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
11	23	34	42	58	74	65
11	23	34	42	58	65	74
11	23	34	42	58	65	74
11	23	34	42	58	65	74

# Sắp xếp nổi bọt (tt) – Mô tả

- Duyệt lần 5

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
11	23	34	42	58	65	74
11	23	34	42	58	65	74
11	23	34	42	58	65	74

- Duyệt lần 6

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
11	23	34	42	58	65	74
11	23	34	42	58	65	74

# Sắp xếp nổi bọt (tt)

- Sau 6 lần duyệt dãy, so sánh và đổi chỗ các cặp phần tử kế tiếp nhau trong dãy ta được dãy sắp xếp theo chiều tăng dần.

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
11	23	34	42	58	65	74

# Sắp xếp nổi bọt (tt)

- Ví dụ 2:

- Cho dãy số sau

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
-35	41	23	15	-62	12	71

- **Yêu cầu:** Minh họa quá trình sắp xếp dãy theo chiều giảm dần dựa vào ý tưởng trên.



# Sắp xếp nổi bọt (tt)

- **Thiết kế giải thuật**

- Thực hiện  $n-1$  lần duyệt dãy (từ trái sang phải):

Sử dụng vòng lặp **for** ( $i=1; i<n-1; i++$ )

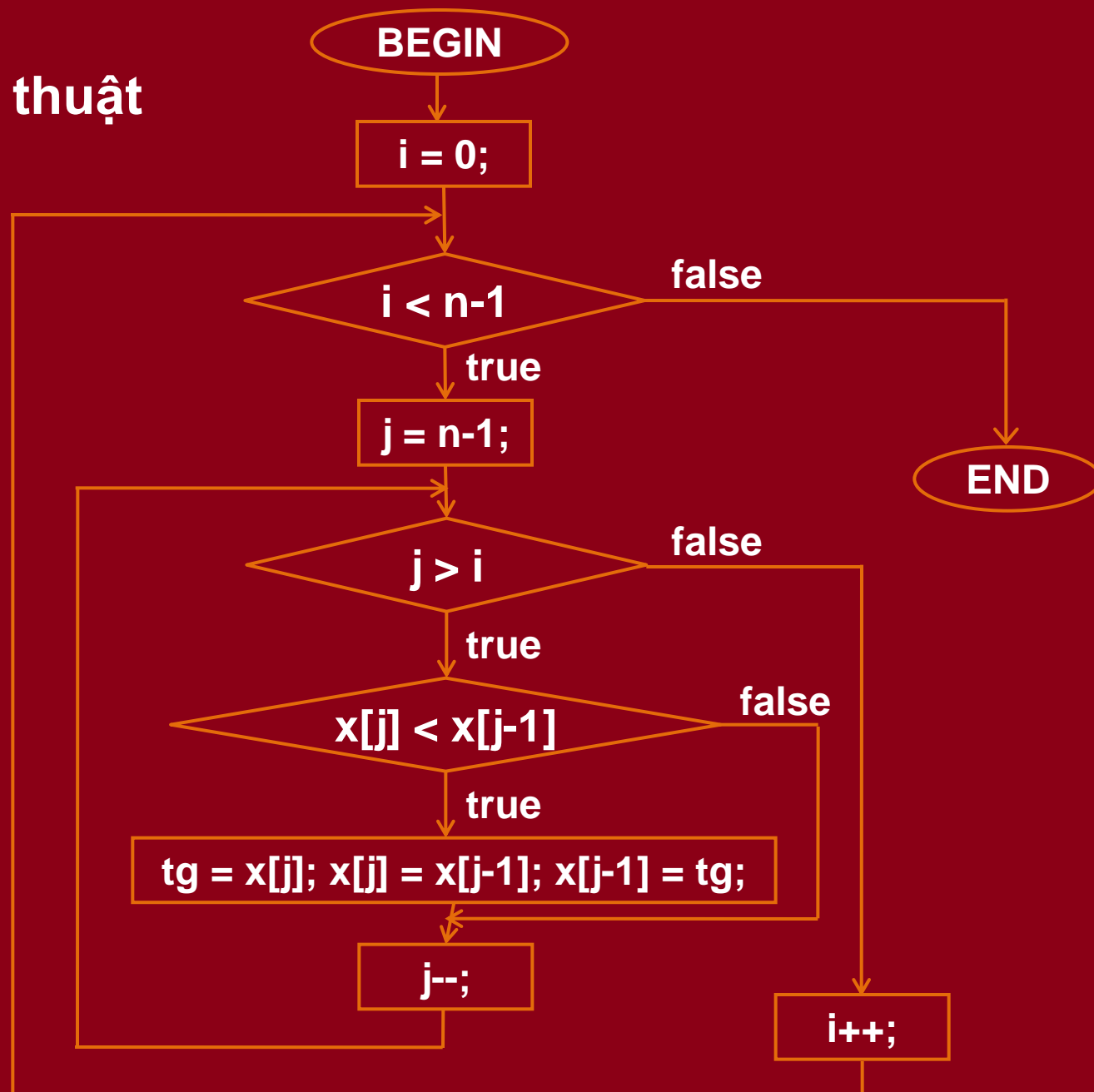
- Mỗi lần duyệt (từ phải sang trái), so sánh các cặp phần tử kế tiếp nhau là  $x[j]$  và  $x[j-1]$ , nếu trái chiều thì đảo giá trị  $x[j]$  và  $x[j-1]$  với ( $j=n-1 \rightarrow i+1$ ):

Duyệt: Sử dụng vòng lặp **for** ( $j=n-1; j>i; j--$ )

So sánh và đổi chỗ: Sử dụng cấu trúc rẽ nhánh

```
if ( $x[j] < x[j-1]$ ) {  
     $tg = x[j]; x[j] = x[j-1]; x[j-1] = tg;$   
}
```

- Lưu đồ giải thuật



# Sắp xếp nổi bọt (tt)

- Giải thuật: Sắp xếp dãy n số nguyên theo chiều tăng dần.

```
void bubbleSort(int x[], int n) {  
    for (int i=0; i<n-1; i++)  
        for (int j=n-1; j>i; j--)  
            if (x[j] < x[j-1])  
            {  
                int tg = x[j];  
                x[j] = x[j-1];  
                x[j-1] = tg;  
            }  
}
```

# Sắp xếp nổi bọt (tt) – Ứng dụng

- Cài đặt ứng dụng thực hiện các yêu cầu:
  - Khởi tạo một số nguyên dương  $n$ .
  - Khởi tạo dãy có  $n$  số nguyên.
  - Sử dụng giải thuật sắp xếp nổi bọt để sắp xếp dãy theo chiều tăng dần.
  - In dãy vừa sắp xếp ra màn hình.

# Sắp xếp nổi bọt (tt) – Bài tập

- **Bài tập 1:**

- a. Trình bày ý tưởng của phương pháp sắp xếp nổi bọt.

- b. Cho dãy số nguyên

**34 74 94 84 54 24**

- Mô tả việc sắp xếp dãy số theo chiều tăng dần (giảm dần) bằng giải thuật sắp xếp nổi bọt.
    - Cài đặt chương trình ứng dụng.

- **Bài tập 2:** Cho dãy từ

**John Wen Thor Ozil Adam Dany**

- Mô tả việc sắp xếp dãy từ theo trật tự từ điển (ngược lại với trật tự từ điển) bằng giải thuật sắp xếp nổi bọt.
  - Cài đặt chương trình ứng dụng.

# Sắp xếp nổi bọt (tt) – Bài tập

- **Bài tập 3: Cài đặt chương trình thực hiện các yêu cầu:**
  - Nhập vào một danh sách học sinh ( $0 < n < 100$ ,  $n$  nhập từ bàn phím), mỗi học sinh gồm các thông tin: Mã học sinh, họ và tên, năm sinh và điểm trung bình.
  - Sắp xếp danh sách theo chiều tăng dần của tên học sinh bằng giải thuật sắp xếp nổi bọt.
  - In danh sách vừa sắp ra màn hình.
  - Sắp xếp danh sách theo chiều giảm dần của điểm trung bình bằng thuật sắp xếp nổi bọt.
  - In danh sách ra màn hình.

## 4.2. Sắp xếp lựa chọn – Selection sort

- Ý tưởng giải thuật

- Dựa vào thuật toán tìm phần tử nhỏ nhất.
- Duyệt và tìm phần tử nhỏ nhất của dãy và đổi chỗ phần tử đó cho phần tử đầu dãy.

- Ví dụ

- Cho dãy số có 5 số nguyên ( $n = 5$ ) được lưu trong mảng  $x$  như dưới đây.

- Dãy  $x$ :

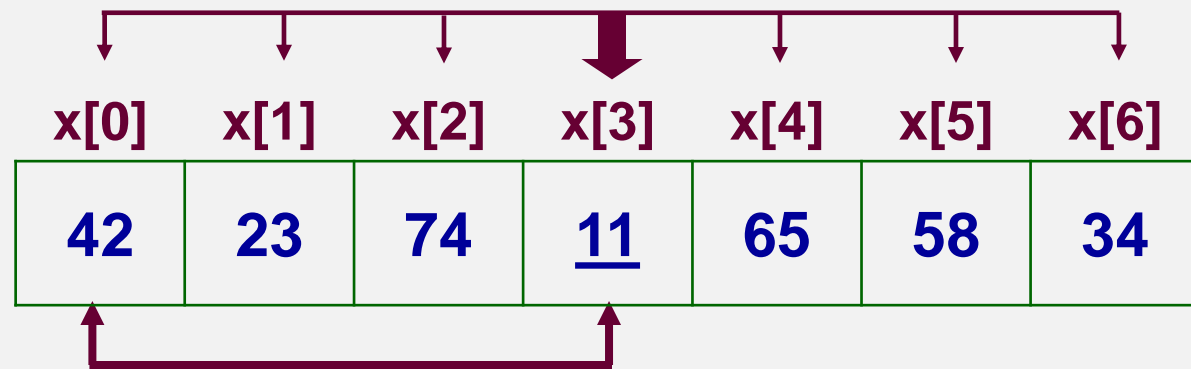
$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$
42	23	74	11	65	58	34

- Yêu cầu: Sắp xếp dãy số theo chiều tăng dần.

# Sắp xếp lựa chọn (tt)

- Thực hiện sắp xếp:

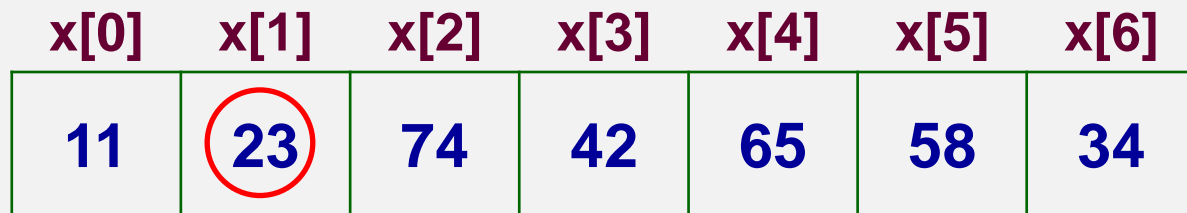
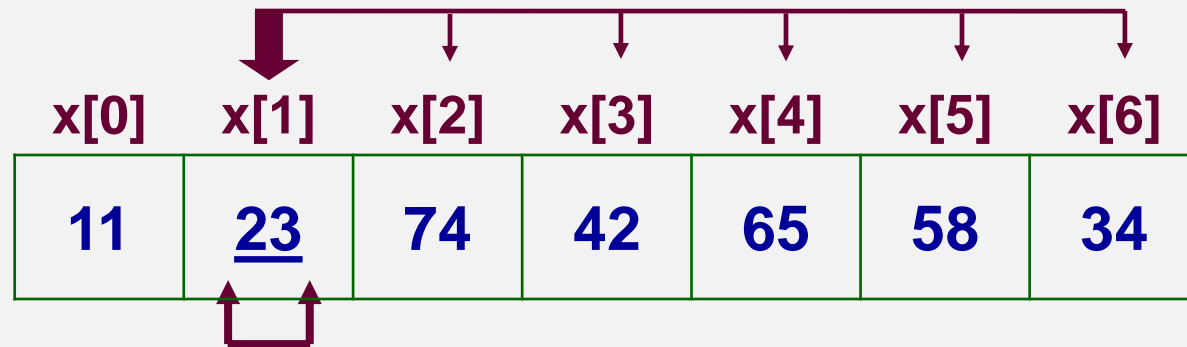
- Duyệt lần 1: Tìm phần tử có giá trị nhỏ nhất trong số các phần tử từ  $x[0]$  đến  $x[6]$ , được  $x[3] = 11$ , đảo giá trị của  $x[3]$  với  $x[0]$ .





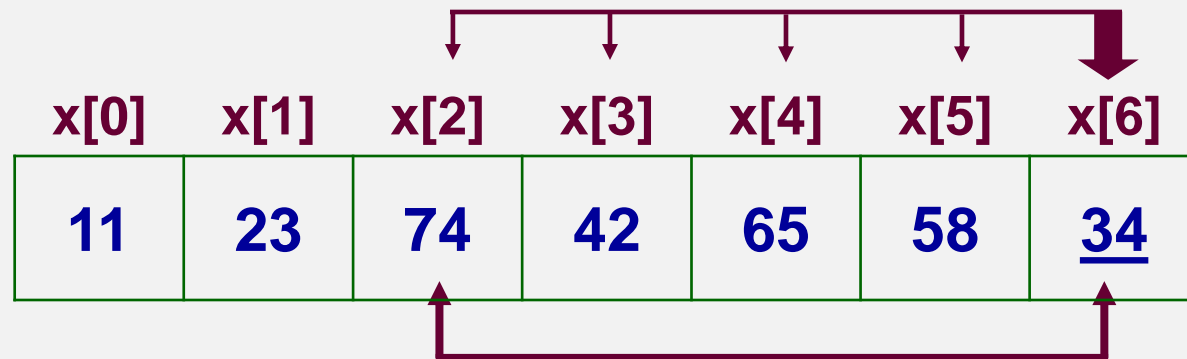
# Sắp xếp lựa chọn (tt)

- Duyệt lần 2:** Tìm phần tử có giá trị nhỏ nhất trong số các phần tử từ  $x[1]$  đến  $x[6]$ , được  $x[1] = 23$ , đảo giá trị của  $x[1]$  với  $x[1]$ .



# Sắp xếp lựa chọn (tt)

- Duyệt lần 3:** Tìm phần tử có giá trị nhỏ nhất trong số các phần tử từ  $x[2]$  đến  $x[6]$ , được  $x[3] = 42$ , đảo giá trị của  $x[3]$  với  $x[2]$ .



# Sắp xếp lựa chọn (tt)

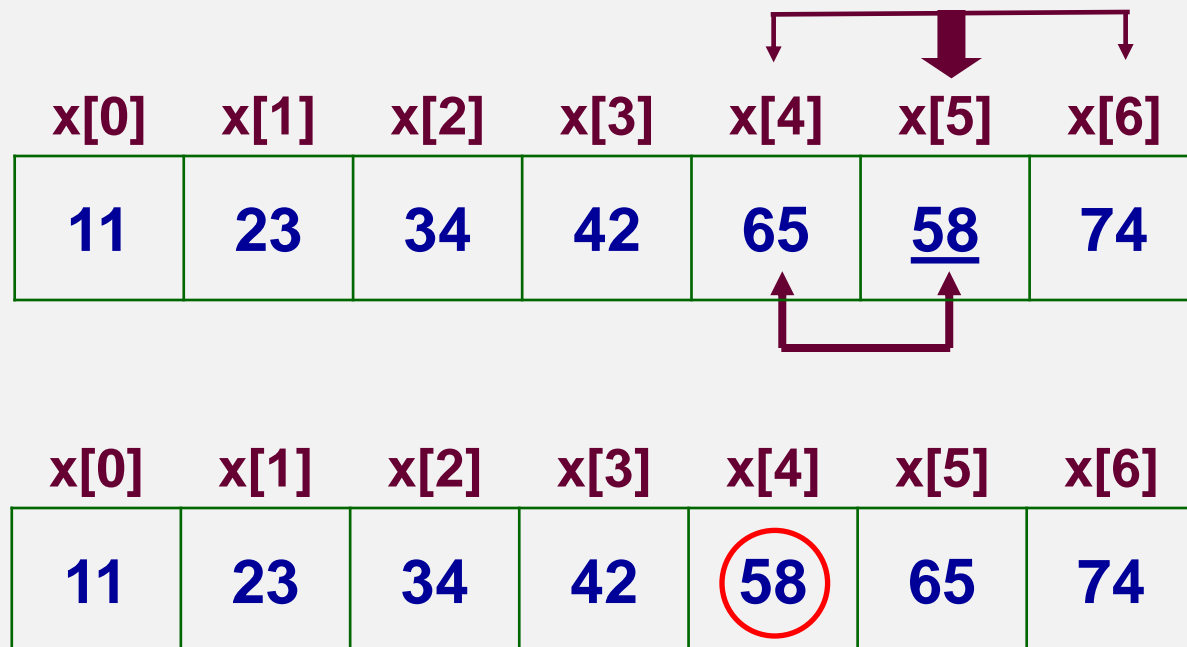
- Duyệt lần 4:** Tìm phần tử có giá trị nhỏ nhất trong số các phần tử từ  $x[3]$  đến  $x[6]$ , được  $x[3] = 42$ , đảo giá trị của  $x[3]$  với  $x[3]$ .

$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$
11	23	34	<u>42</u>	65	58	74

$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$
11	23	34	42	65	58	74

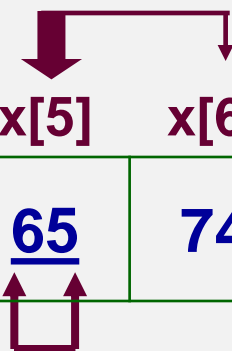
# Sắp xếp lựa chọn (tt)

- Duyệt lần 5:** Tìm phần tử có giá trị nhỏ nhất trong số các phần tử từ  $x[4]$  đến  $x[6]$ , được  $x[5] = 58$ , đảo giá trị của  $x[5]$  với  $x[4]$ .



# Sắp xếp lựa chọn (tt)

- Duyệt lần 6:** Tìm phần tử có giá trị nhỏ nhất trong số các phần tử từ  $x[5]$  đến  $x[6]$ , được  $x[5] = 65$ , đảo giá trị của  $x[5]$  với  $x[5]$ .



$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$
11	23	34	42	58	<u>65</u>	74

$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$
11	23	34	42	58	65	74

# Sắp xếp lựa chọn (tt)

- Ví dụ áp dụng:
  - Cho dãy số nguyên như sau:

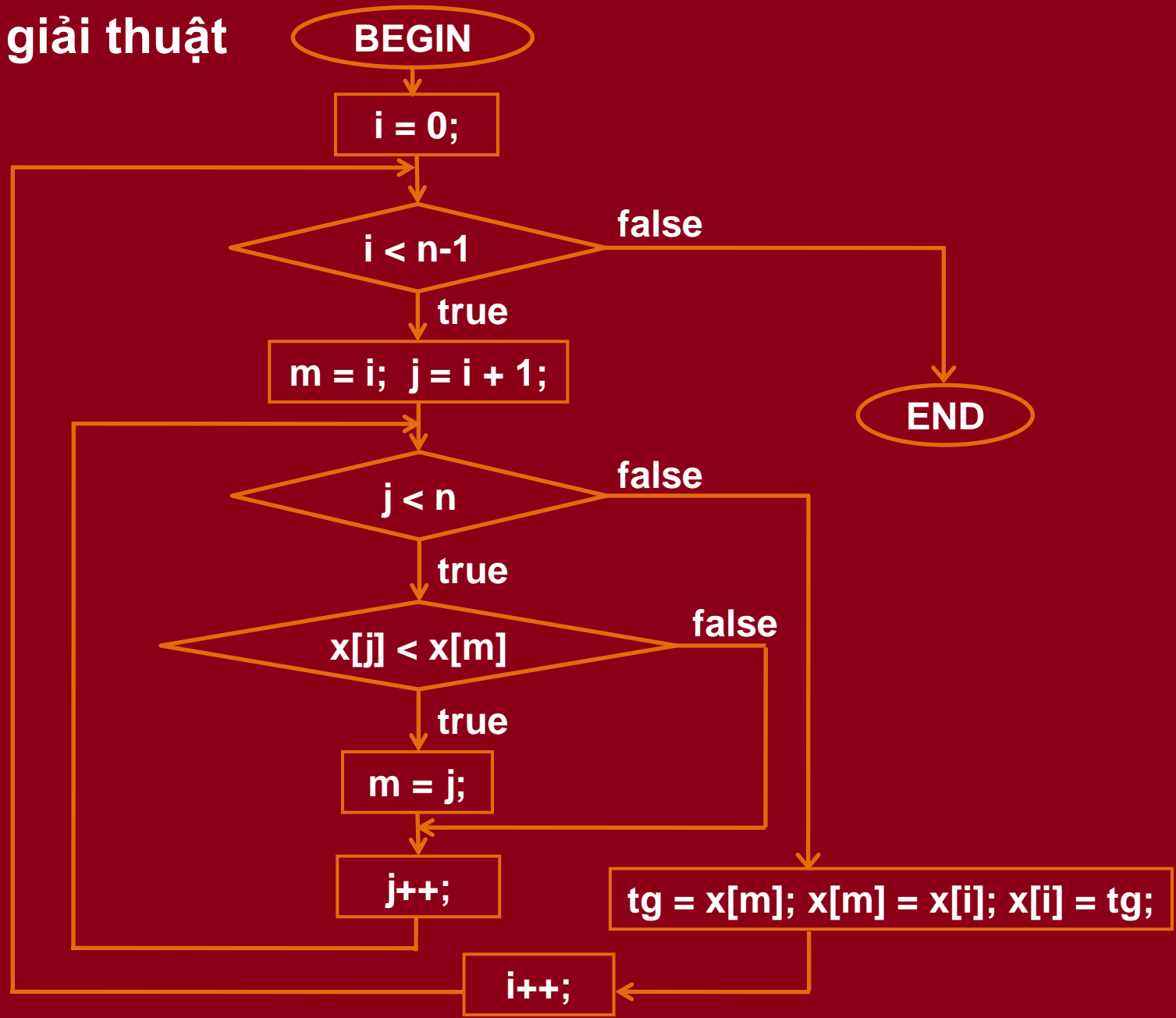
x[0]	x[1]	x[2]	x[3]	x[4]	x[5]
53	-21	67	15	82	-14

- **Yêu cầu:** Dựa vào ý tưởng trên, minh họa việc sắp xếp dãy số theo chiều giảm dần.

# Sắp xếp lựa chọn (tt)

- **Thiết kế giải thuật**
  - Thực hiện  $n - 1$  lần duyệt dãy (từ trái sang phải).
  - Mỗi lần duyệt tìm phần tử nhỏ nhất trong số các phần tử từ  $x[i]$  đến  $x[n-1]$ , giả sử là  $x[m]$ , đổi chỗ  $x[m]$  và  $x[i]$  (với  $i = [0, n-2]$ )

■ Lưu đồ giải thuật





# Sắp xếp lựa chọn (tt)

- Giải thuật (sắp tăng dần)

```
void selectionSort(int X[], int n) {  
    for (int i=0; i<n-1; i++) {  
        int m = i;  
        for (int j=i+1; j<n; j++)  
            if (X[j] < X[m])  
                m = j;  
        int tg = X[m];  
        X[m] = X[i];  
        X[i] = tg;  
    }  
}
```

# Sắp xếp lựa chọn (tt) – Ứng dụng

- Cài đặt ứng dụng thực hiện các yêu cầu:
  - Khởi tạo một số nguyên dương  $n$ .
  - Khởi tạo dãy có  $n$  số nguyên.
  - Sắp xếp dãy theo chiều tăng dần bằng giải thuật sắp xếp lựa chọn.
  - In dãy vừa sắp ra màn hình.

# Sắp xếp lựa chọn (tt) – Bài tập

- **Bài tập 1:** Cho dãy số nguyên

74 34 24 84 54 94 14 64 04

- Minh họa việc sắp xếp dãy số theo chiều tăng dần (giảm dần) bằng phương pháp sắp xếp lựa chọn.
- Cài đặt chương trình sắp xếp dãy số.

- **Bài tập 2:** Cho dãy từ

John Wen Anna Henry Thor Terry Ozil Adam Dany

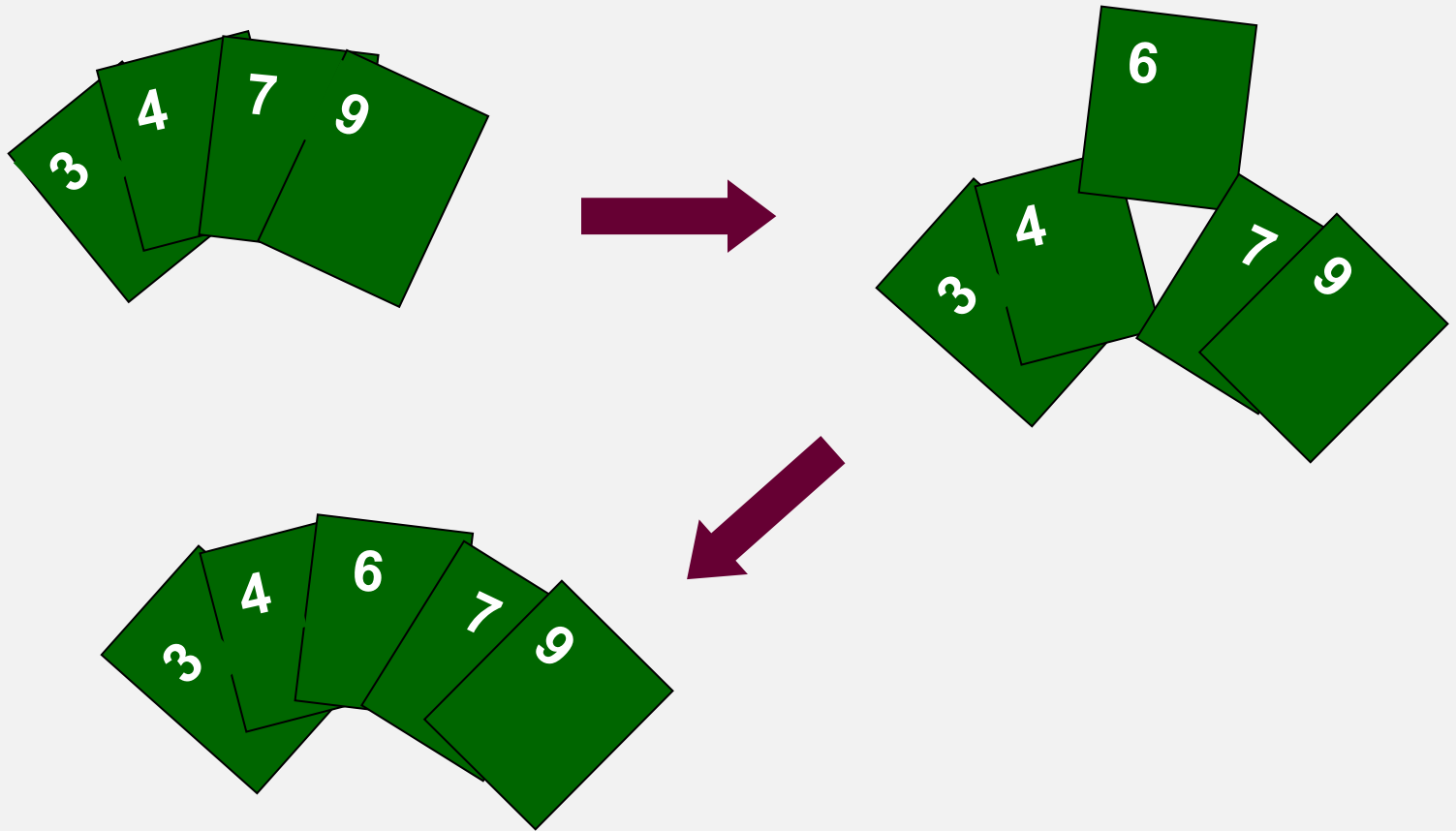
- Minh họa việc sắp xếp dãy từ theo trật tự từ điển (ngược lại với trật tự từ điển) bằng phương pháp sắp xếp lựa chọn.
- Cài đặt chương trình sắp xếp dãy từ.

# Sắp xếp lựa chọn (tt) – Bài tập

- **Bài tập 3: Cài đặt chương trình thực hiện các yêu cầu:**
  - Nhập vào một danh sách học sinh ( $0 < n < 100$ ,  $n$  nhập từ bàn phím), mỗi học sinh gồm các thông tin: Mã học sinh, họ và tên, năm sinh và điểm trung bình.
  - Sắp xếp danh sách theo chiều tăng dần của tên học sinh bằng giải thuật sắp xếp lựa chọn.
  - In danh sách vừa sắp ra màn hình.
  - Sắp xếp danh sách theo chiều giảm dần của điểm trung bình bằng thuật sắp xếp lựa chọn.
  - In danh sách ra màn hình.

## 4.3. Sắp xếp chèn – Insertion sort

- Bài toán gợi ý tưởng:



# Sắp xếp chèn (tt)

- Ý tưởng giải thuật

- Chèn một phần tử vào dãy đã được sắp xếp sao cho trật tự của dãy không bị thay đổi..
- Duyệt và tìm vị trí thích hợp cho phần tử mới trong dãy.
- Chèn phần tử mới vào vị trí thích hợp.

- Ví dụ

- Cho dãy số có 5 số nguyên ( $n = 5$ ) được lưu trong mảng x như dưới đây.

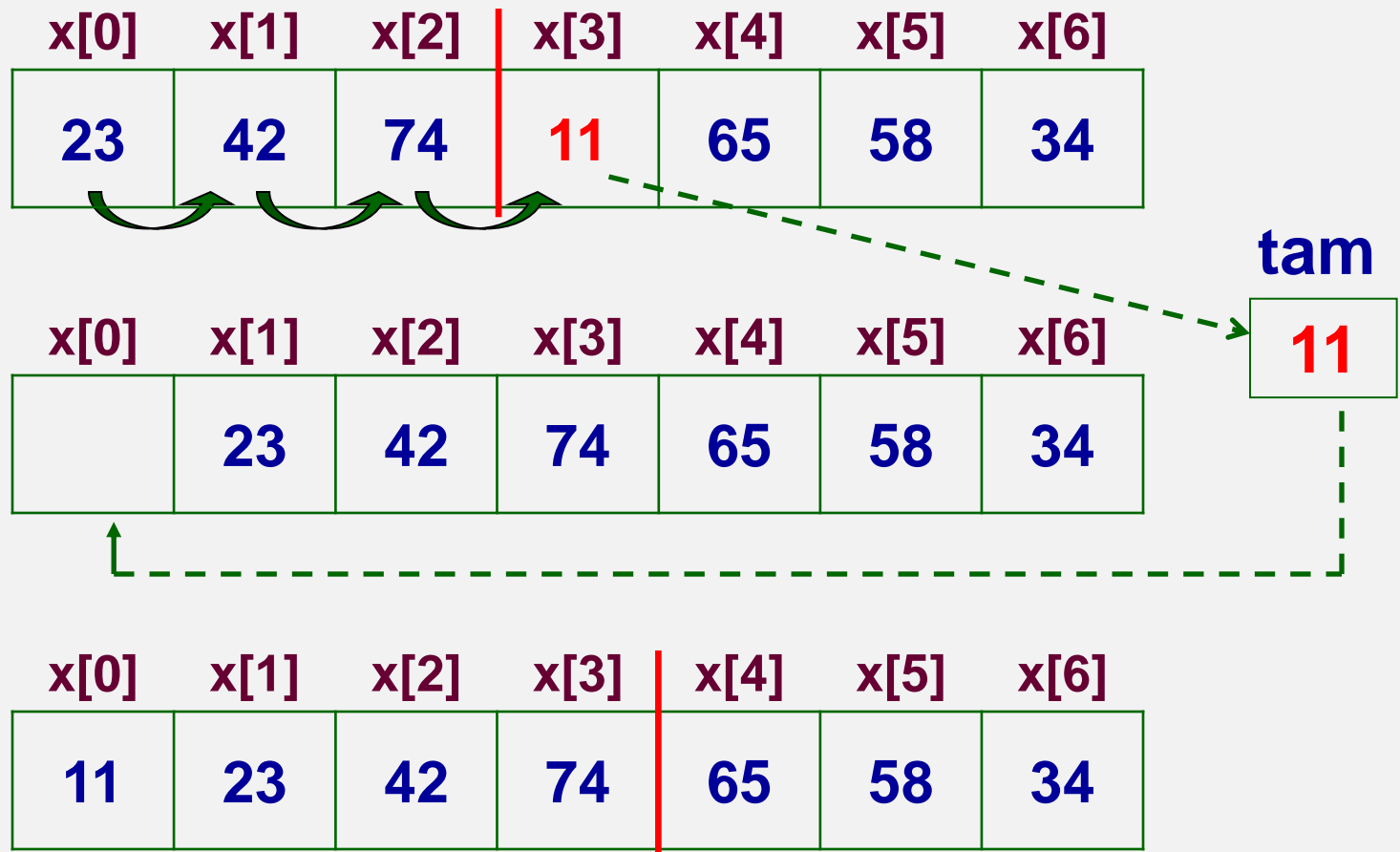
- Dãy x:

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]
42	23	74	11	65	58	34

- Yêu cầu: Sắp xếp dãy số theo chiều tăng dần.

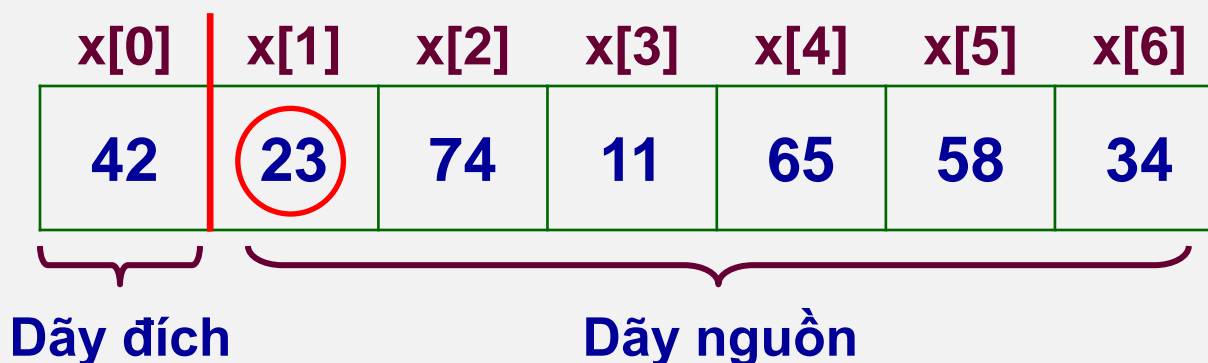
# Sắp xếp chèn (tt)

- **Thể hiện ý tưởng:** Chèn số **11** vào phía trước đã sắp xếp theo chiều tăng dần.



# Sắp xếp chèn (tt)

- **Đưa ý tưởng vào bài toán sắp xếp:**
  - Xem dãy cần sắp gồm 2 dãy nối tiếp.
  - Dãy trái (dãy đích) gồm các phần tử đã được sắp xếp, dãy phải (dãy nguồn) là các phần tử chưa được sắp xếp.

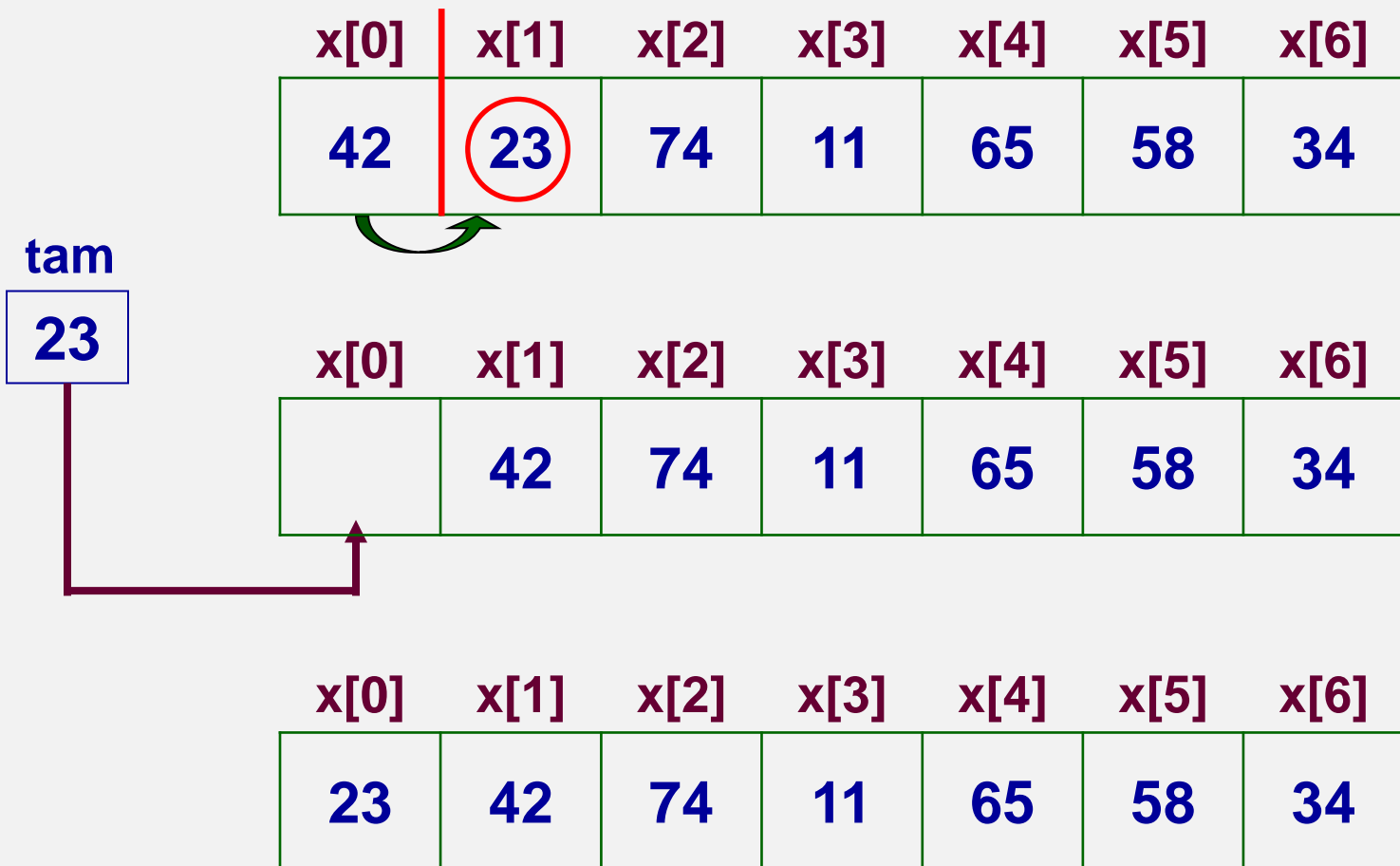


- Lấy phần tử đầu dãy nguồn chèn vào vị trí thích hợp trong dãy đích.



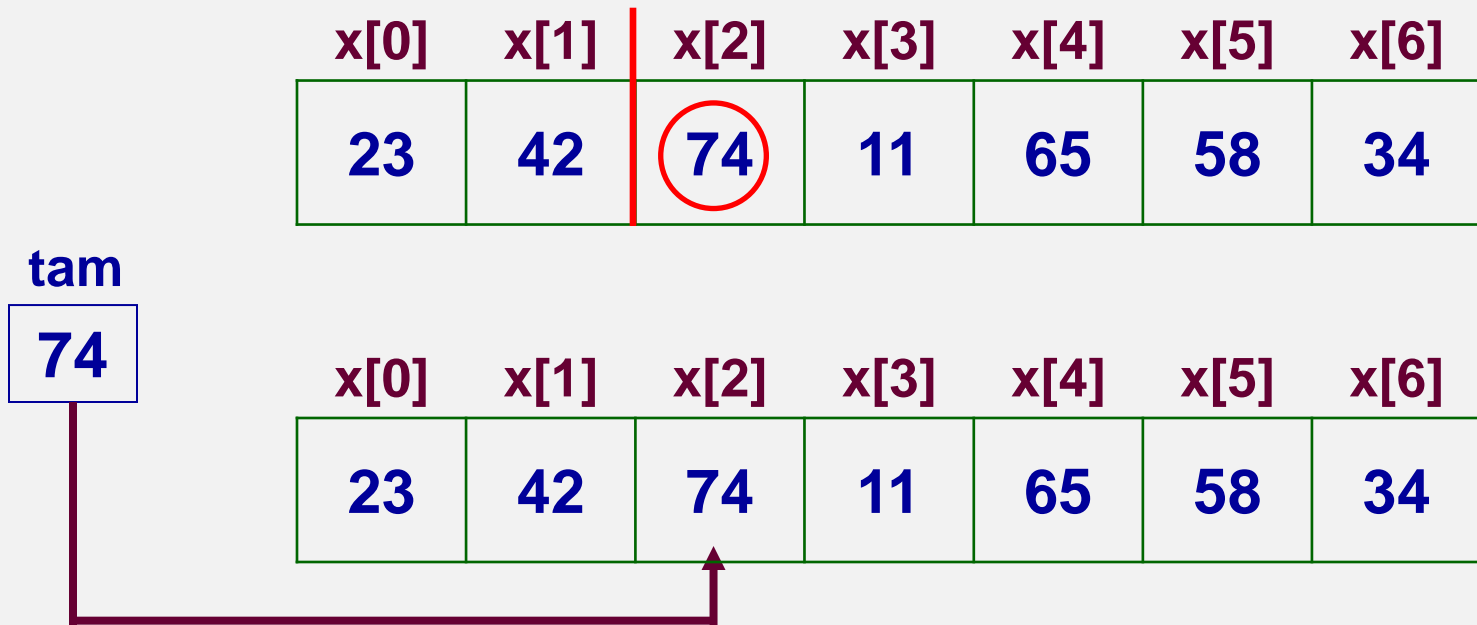
# Sắp xếp chèn (tt)

- Lượt chia 1: Chia ở vị trí  $x[1]$ ,  $\text{tam} = x[1] = 23$ .



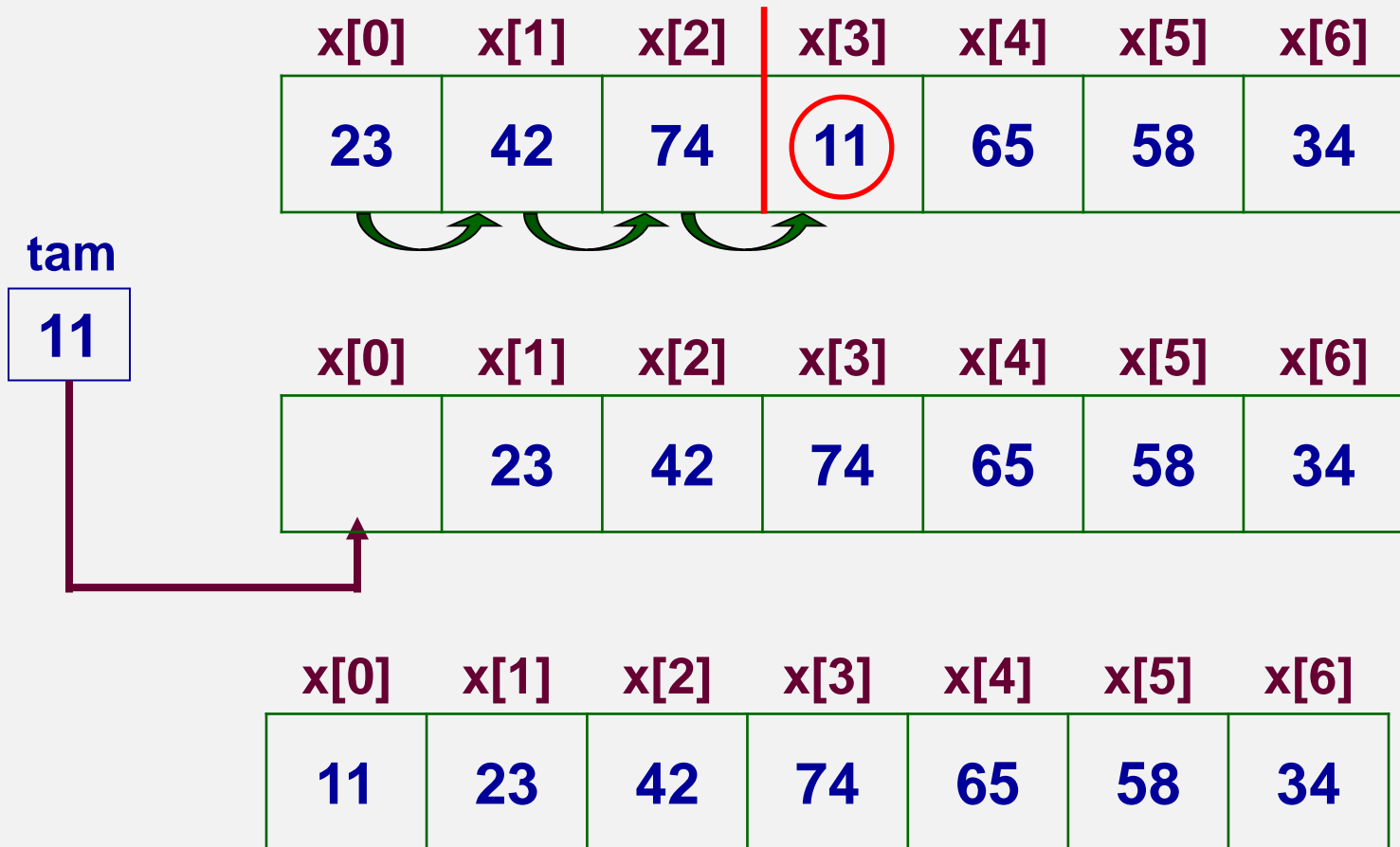
# Sắp xếp chèn (tt)

- Lượt chia 2: Chia ở vị trí  $x[2]$ ,  $\text{tam} = x[2] = 74$ .



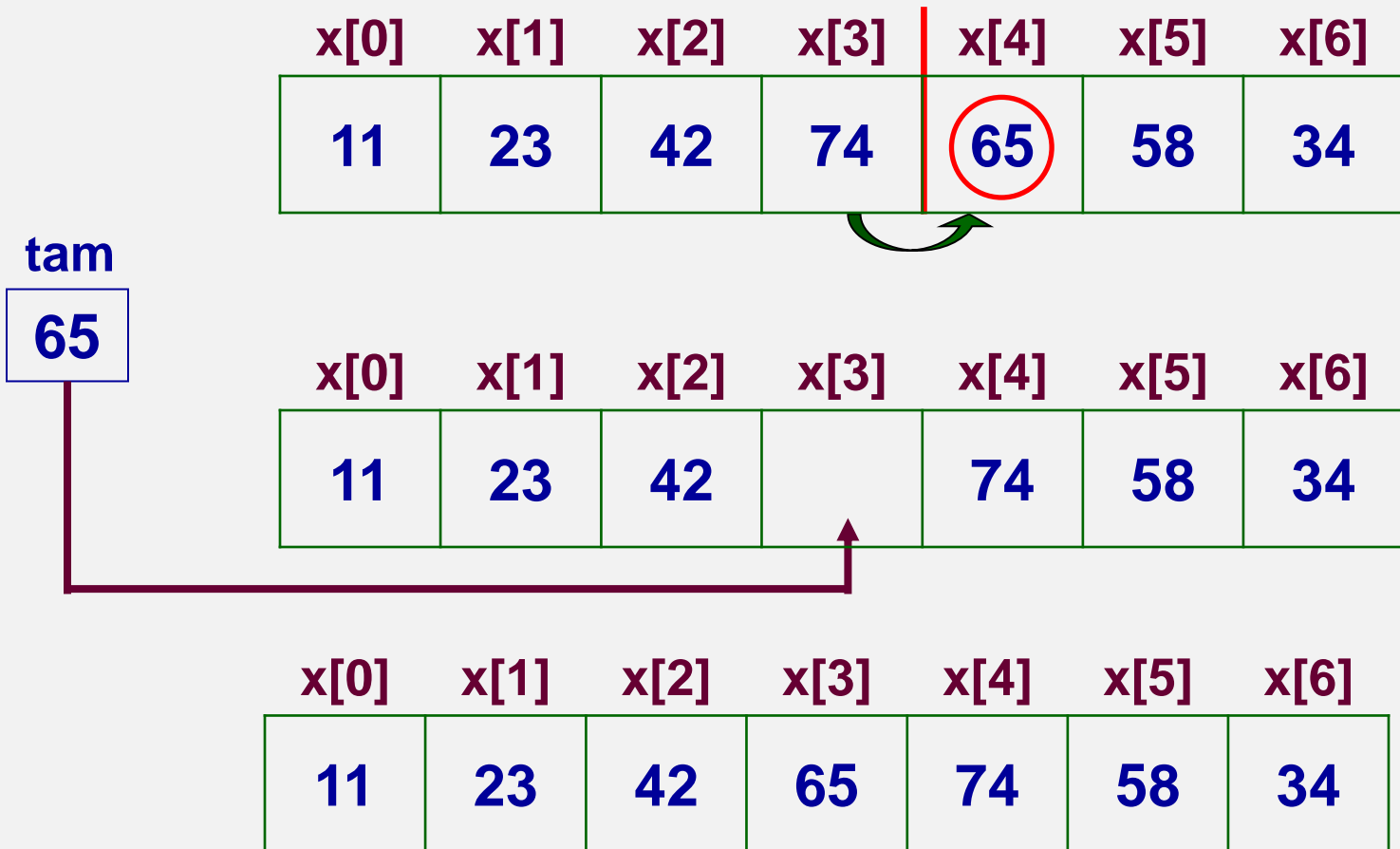
# Sắp xếp chèn (tt)

- Lượt chia 3: Chia ở vị trí  $x[3]$ ,  $\text{tam} = x[3] = 11$ .



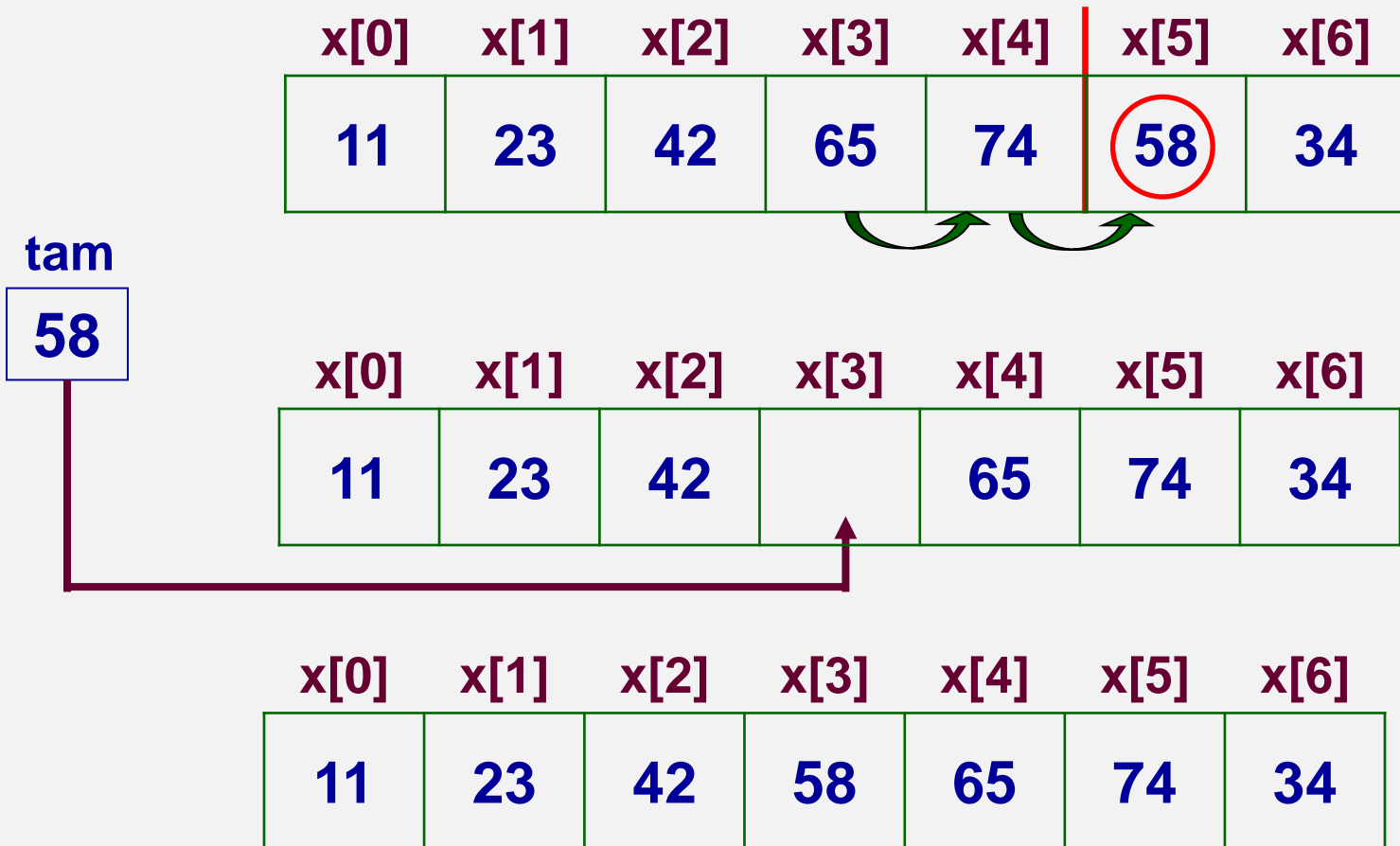
# Sắp xếp chèn (tt)

- Lượt chia 4: Chia ở vị trí  $x[4]$ ,  $\text{tam} = x[4] = 65$ .



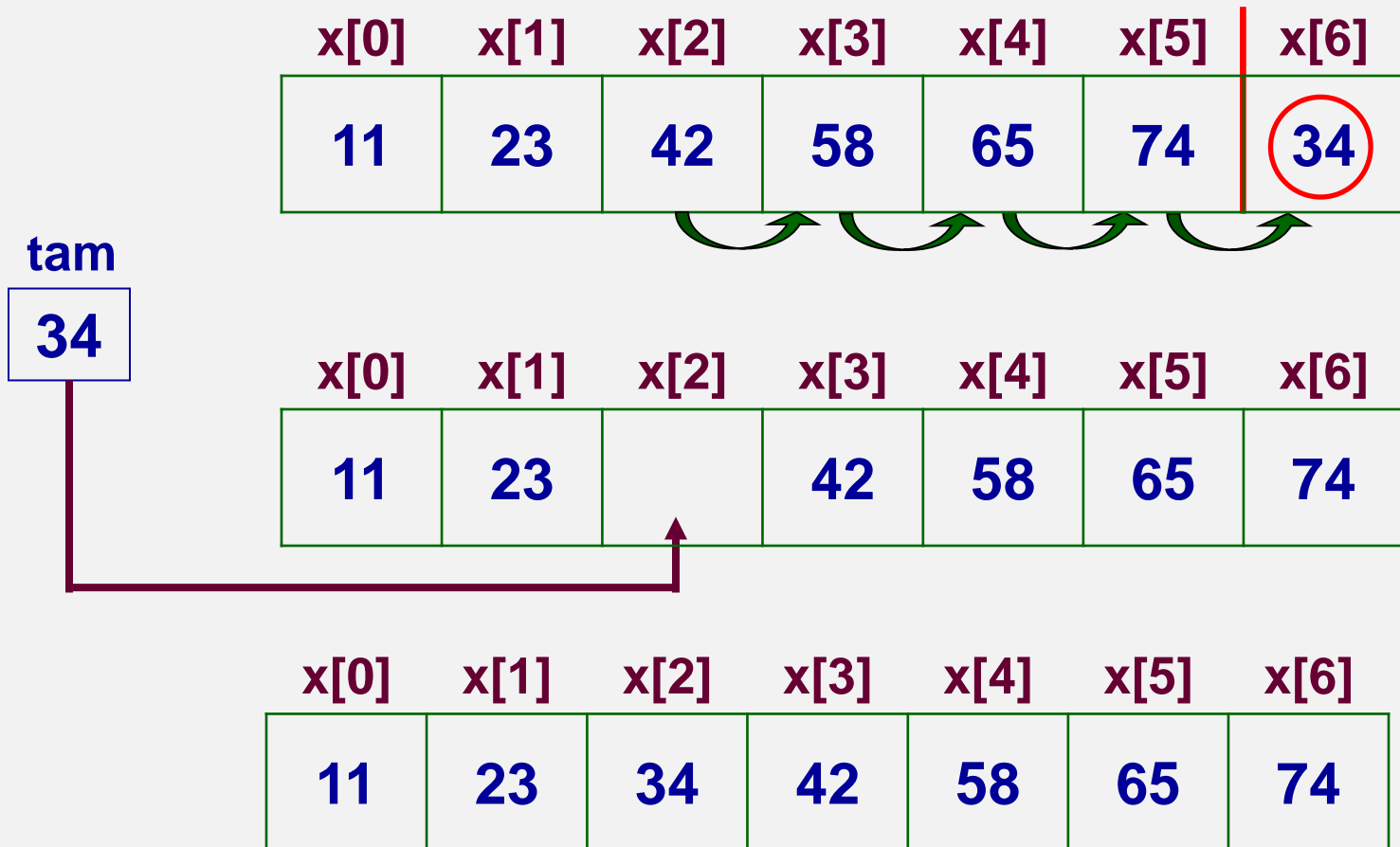
# Sắp xếp chèn (tt)

- Lượt chia 5: Chia ở vị trí  $x[5]$ ,  $\text{tam} = x[5] = 58$ .



# Sắp xếp chèn (tt)

- Lượt chia 6: Chia ở vị trí  $x[6]$ ,  $\text{tam} = x[6] = 34$ .



# Sắp xếp chèn (tt)

- Ví dụ 2

- Cho dãy số nguyên như sau:

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]
53	-21	67	15	60	40

- Yêu cầu:

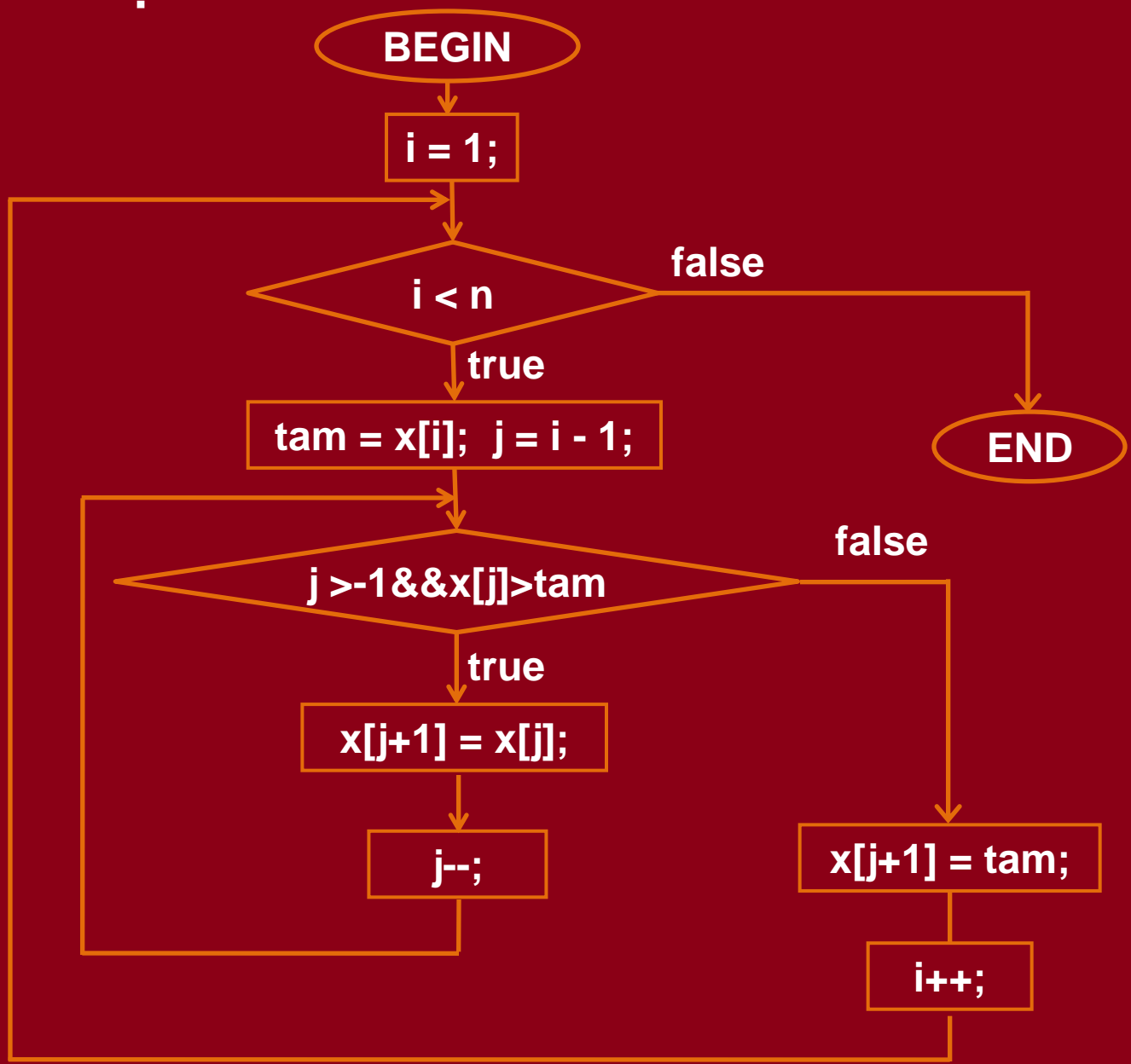
- Dựa vào ý tưởng trên, minh họa việc sắp xếp dãy số theo chiều tăng dần.
- Dựa vào ý tưởng trên, minh họa việc sắp xếp dãy số theo chiều giảm dần.

# Sắp xếp chèn (tt)

- **Thiết kế giải thuật**
  - Thực hiện  $n-1$  lần “chia” dãy thành dãy đích và dãy nguồn:
    - Dãy đích gồm các phần tử  $x[0]$  đến  $x[i-1]$  ( $i = 1 \rightarrow n-1$ )
    - Dãy nguồn gồm các phần tử  $x[i]$  đến  $x[n-1]$
  - Mỗi lần chia lấy phần tử đầu của dãy nguồn (là  $x[i]$  với  $i = 1 \rightarrow n-1$ ), chèn vào vị trí thích hợp trong dãy đích.



■ Lưu đồ giải thuật



# Sắp xếp chèn (tt)

- Giải thuật

```
void insertionSort(int x[], int n) {  
    for (int i=1; i<n; i++) {  
        int tam = x[i];  
        int j = i-1;  
        while (j>-1 && x[j]>tam) {  
            x[j+1] = x[j];  
            j--;  
        }  
        x[j+1] = tam;  
    }  
}
```

# Sắp xếp chèn (tt) – Ứng dụng

- Cài đặt ứng dụng thực hiện các yêu cầu:
  - Khởi tạo một số nguyên dương  $n$ .
  - Khởi tạo dãy có  $n$  số nguyên.
  - Sắp xếp dãy theo chiều tăng dần bằng giải thuật sắp xếp chèn.
  - In dãy vừa sắp ra màn hình.

# Sắp xếp chèn (tt) – Bài tập

## ■ Bài tập 1: Cho dãy số nguyên

74 34 24 84 54 94 14 64 04

- Minh họa việc sắp xếp dãy số theo chiều tăng dần (giảm dần) bằng phương pháp sắp xếp chèn.
- Cài đặt chương trình sắp xếp dãy số.

## ■ Bài tập 2: Cho dãy từ

John Wen Anna Henry Thor Terry Ozil Adam Dany

- Minh họa việc sắp xếp dãy từ theo trật tự từ điển (ngược lại với trật tự từ điển) bằng phương pháp sắp xếp chèn.
- Cài đặt chương trình sắp xếp dãy từ.

# Sắp xếp chèn (tt) – Bài tập

- **Bài tập 3: Cài đặt chương trình thực hiện các yêu cầu:**
  - Nhập vào một danh sách học sinh ( $0 < n < 100$ ,  $n$  nhập từ bàn phím), mỗi học sinh gồm các thông tin: Mã học sinh, họ và tên, năm sinh và điểm trung bình.
  - Sắp xếp danh sách theo chiều tăng dần của tên học sinh bằng giải thuật sắp xếp chèn.
  - In danh sách vừa sắp ra màn hình.
  - Sắp xếp danh sách theo chiều giảm dần của điểm trung bình bằng thuật sắp xếp chèn.
  - In danh sách ra màn hình.