**Homework lecture 1**

**Sort**

1. Given a list of integer numbers: 2, 1, 10, 6, 3, 8, 7, 13, 20. Draw the steps to sort (ascending sort) the list by following methods:

* Selection sort

2 1 10 6 3 8 7 13 20

1 2 10 6 3 8 7 13 20

1 2 10 6 3 8 7 13 20

1 2 3 6 10 8 7 13 20

1 2 3 6 7 8 10 13 20

1 2 3 6 7 8 10 13 20

1 2 3 6 7 8 10 13 20

1 2 3 6 7 8 10 13 20

1 2 3 6 7 8 10 13 20

* Insertion sort

2 1 10 6 3 8 7 13 20

1 2 10 6 3 8 7 13 20

1 2 10 6 3 8 7 13 20

1 2 6 10 3 8 7 13 20

1 2 3 6 10 8 7 13 20

1 2 3 6 8 10 7 13 20

1 2 3 6 7 8 10 13 20

1 2 3 6 7 8 10 13 20

1 2 3 6 7 8 10 13 20

* Bubble sort

2 1 10 6 3 8 7 13 20

Loop 1: 1 2 10 6 3 8 7 13 20

1 2 6 10 3 8 7 13 20

1 2 6 3 10 8 7 13 20

1 2 6 3 8 10 7 13 20

1 2 6 3 8 7 10 13 20

Loop 2: 1 2 3 6 8 7 10 13 20

1 2 3 6 7 8 10 13 20

* Merge sort

2 1 10 6 3 8 7 13 20

2 1 10 6 3 8 7 13 20

2 1 10 6 3 8 7 13 20

2 1 10 6 3 8 7 13 20

2 1

1 2

1 2 10 3 6 7 8 13 20

1 2 3 6 10 7 8 13 20

1 2 3 6 7 8 10 13 20

* Quick sort

2 1 10 6 3 8 7 13 20

2 1 10 6 3 8 7 13 20(pivot)

2 1 10 6 3 8 7 13(pivot) 20

2 1 6 3 7(pivot) 10 8 13 20

2 1 3(pivot) 6 7 8(pivot) 10 13 20

1(pivot) 2 3 6 7 8 10 13 20

* Heap sort

2 1 10 6 3 8 10 13 20

2 2 10 10 10 10

1 1 10 1 2 1 2 6 2 6 2

6 1 1 3

10 10 10 10

6 2 6 8 6 8 6 10

1 3 8 1 3 2 1 3 2 10 1 3 2 8

10 10 20

6 10 6 10 13 10

1 3 2 8 13 3 2 8 🡺 10 3 2 8

13 1 1 6

6 13 13

13 10 6 10 10 10

10 3 2 8 10 3 2 8 6 3 2 8

1 1 1

(lấy phần tử dưới cùng bên phải thay cho gốc, gốc đưa về cuối mảng) đổi vị trí gốc mới sao cho vẫn được cây max-heap và tiếp tục lấy gốc ra khỏi cây nhị phân

🡪1 2 3 6 7 8 10 13 20

* Radix sort

2 1 10 6 3 8 7 13 20

10 20 1 2 3 13 6 7 8

1 2 3 6 7 8 10 13 20

1. Same question for these lists:

* 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
* Selection sort

1 2 3 4 5 6 7 8 9 10

* Insertion sort

1 2 3 4 5 6 7 8 9 10

* Bubble sort

1 2 3 4 5 6 7 8 9 10

* Merge sort

1 2 3 4 5 6 7 8 9 10

1 2 3 4 5 6 7 8 9 10

1 2 3 4 5 6 7 8 9 10

1 2 3 4 5 6 7 8 9 10

1 2 45 6 7 9 10

1 2 67

1 2 3 6 7 8

1 2 3 4 5 6 7 8 9 10

1 2 3 4 5 6 7 8 9 10

* Quick sort

1 2 3 4 5 6 7 8 9 10 (pivot)

1 2 3 4 5 6 7 8 9 (pivot) 10

………

1 2 3 4 5 6 7 8 9 10

* Heap sort
* Radix sort

1 2 3 4 5 6 7 8 9 10

10 1 2 3 4 5 6 7 8 9

1 2 3 4 5 6 7 8 9 10

* 10, 9, 8, 7, 6, 5, 4, 3, 2, 1
* Selection sort

10 9 8 7 6 5 4 3 2 1

1 9 8 7 6 5 4 3 2 10

1 2 8 7 6 5 4 3 9 10

1 2 3 7 6 5 4 8 9 10

1 2 3 4 6 5 7 8 9 10

1 2 3 4 5 6 7 8 9 10

* Insertion sort

10 9 8 7 6 5 4 3 2 1

9 10 8 7 6 5 4 3 2 1

8 9 10 7 6 5 4 3 2 1

7 8 9 10 6 5 4 3 2 1

6 7 8 9 10 5 4 3 2 1

5 6 7 8 9 10 4 3 2 1

4 5 6 7 8 9 10 3 2 1

3 4 5 6 7 8 9 10 2 1

2 3 4 5 6 7 8 9 10 1

1 2 3 4 5 6 7 8 9 10

* Bubble sort

10 9 8 7 6 5 4 3 2 1

Loop 1:

9 10 8 7 6 5 4 3 2 1

9 8 10 7 6 5 4 3 2 1

9 8 7 10 6 5 4 3 2 1

9 8 7 6 10 5 4 3 2 1

9 8 7 6 5 10 4 3 2 1

9 8 7 6 5 4 10 3 2 1

9 8 7 6 5 4 3 10 2 1

9 8 7 6 5 4 3 2 10 1

9 8 7 6 5 4 3 2 1 10

Loop 2:

8 9 7 6 5 4 3 2 1 10

8 7 9 6 5 4 3 2 1 10

8 7 6 9 5 4 3 2 1 10

8 7 6 5 9 4 3 2 1 10

8 7 6 5 4 9 3 2 1 10

8 7 6 5 4 3 9 2 1 10

8 7 6 5 4 3 2 9 1 10

8 7 6 5 4 3 2 1 9 10

Loop 3:

7 8 6 5 4 3 2 1 9 10

7 6 8 5 4 3 2 1 9 10

7 6 5 8 4 3 2 1 9 10

7 6 5 4 8 3 2 1 9 10

7 6 5 4 3 8 2 1 9 10

7 6 5 4 3 2 8 1 9 10

7 6 5 4 3 2 1 8 9 10

Loop 4:

6 7 5 4 3 2 1 8 9 10

6 5 7 4 3 2 1 8 9 10

6 5 4 7 3 2 1 8 9 10

6 5 4 3 7 2 1 8 9 10

6 5 4 3 2 7 1 8 9 10

6 5 4 3 2 1 7 8 9 10

Loop 5:

5 6 4 3 2 1 7 8 9 10

5 4 6 3 2 1 7 8 9 10

5 4 3 6 2 1 7 8 9 10

5 4 3 2 6 1 7 8 9 10

5 4 3 2 1 6 7 8 9 10

Loop 6:

4 5 3 2 1 6 7 8 9 10

4 3 5 2 1 6 7 8 9 10

4 3 2 5 1 6 7 8 9 10

4 3 2 1 5 6 7 8 9 10

Loop 7:

3 4 2 1 5 6 7 8 9 10

3 2 4 1 5 6 7 8 9 10

3 2 1 4 5 6 7 8 9 10

Loop 8:

2 3 1 4 5 6 7 8 9 10

2 1 3 4 5 6 7 8 9 10

Loop 9:

1 2 3 4 5 6 7 8 9 10

* Merge sort

10 9 8 7 6 5 4 3 2 1

10 9 8 7 6 5 4 3 2 1

10 9 8 7 6 5 4 3 2 1

10 9 8 7 6 5 4 3 2 1

10 9 5 4

9 10 6 7 4 5 1 2

8 9 10 3 4 5

6 7 8 9 10 1 2 3 4 5

1 2 3 4 5 6 7 8 9 10

* Quick sort

10 9 8 7 6 5 4 3 2 1

1(pivot) 10 9 8 7 6 5 4 3 2

1 2(pivot) 10 9 8 7 6 5 4 3 2

.......

1 2 3 4 5 6 7 8 9 10

* Heap sort
* Radix sort

10 9 8 7 6 5 4 3 2 1

10 1 2 3 4 5 6 7 8 9

1 2 3 4 5 6 7 8 9 10

* 9, 10, 1, 2, 3, 4, 5, 6, 7, 8
* Selection sort

9 10 1 2 3 4 5 6 7 8

1 10 9 2 3 4 5 6 7 8

1 2 9 10 3 4 5 6 7 8

1 2 3 10 9 4 5 6 7 8

1 2 3 4 9 10 5 6 7 8

1 2 3 4 5 10 9 6 7 8

1 2 3 4 5 6 9 10 7 8

1 2 3 4 5 6 7 10 9 8

1 2 3 4 5 6 7 8 9 10

1 2 3 4 5 6 7 8 9 10

* Insertion sort

9 10 1 2 3 4 5 6 7 8

9 10 1 2 3 4 5 6 7 8

1 9 10 2 3 4 5 6 7 8

1 2 9 10 3 4 5 6 7 8

1 2 3 9 10 4 5 6 7 8

1 2 3 4 9 10 5 6 7 8

1 2 3 4 5 9 10 6 7 8

1 2 3 4 5 6 9 10 7 8

1 2 3 4 5 6 7 9 10 8

1 2 3 4 5 6 7 8 9 10

* Bubble sort

9 10 1 2 3 4 5 6 7 8

Loop 1:

9 1 10 2 3 4 5 6 7 8

9 1 2 10 3 4 5 6 7 8

9 1 2 3 10 4 5 6 7 8

9 1 2 3 4 10 5 6 7 8

9 1 2 3 4 5 10 6 7 8

9 1 2 3 4 5 6 10 7 8

9 1 2 3 4 5 6 7 10 8

9 1 2 3 4 5 6 7 8 10

Loop 2:

1 9 2 3 4 5 6 7 8 10

1 2 9 3 4 5 6 7 8 10

1 2 3 9 4 5 6 7 8 10

1 2 3 4 9 5 6 7 8 10

1 2 3 4 5 9 6 7 8 10

1 2 3 4 5 6 9 7 8 10

1 2 3 4 5 6 7 9 8 10

1 2 3 4 5 6 7 8 9 10

* Merge sort

9 10 1 2 3 4 5 6 7 8

9 10 1 2 3 4 5 6 7 8

9 10 1 2 3 4 5 6 7 8

9 10 1 2 3 4 5 6 7 8

9 10 4 5

9 10 2 3 4 5 7 8

1 9 10 4 5 6

1 2 3 9 10 4 5 6 7 8

1 2 3 4 5 6 7 8 9 10

* Quick sort

9 10 1 2 3 4 5 6 7 8

1 2 3 4 5 6 7 8(pivot) 9 10

1 2 3 4 5 6 7(pivot) 8 9 10(pivot)

1 2 3 4 5 6(pivot) 7 8 9 10

1 2 3 4 5(pivot) 6 7 8 9 10

1 2 3 4(pivot) 5 6 7 8 9 10

1 2 3(pivot) 4 5 6 7 8 9 10

1 2(pivot) 3 4 5 6 7 8 9 10

* Heap sort
* Radix sort

9 10 1 2 3 4 5 6 7 8

10 1 2 3 4 5 6 7 8 9

1 2 3 4 5 6 7 8 9 10

* 3, 4, 5, 6, 7, 8, 9, 10, 1, 2
* Selection sort

3 4 5 6 7 8 9 10 1 2

1 4 5 6 7 8 9 10 3 2

1 2 5 6 7 8 9 10 3 4

1 2 3 6 7 8 9 10 5 4

1 2 3 4 7 8 9 10 5 6

1 2 3 4 5 8 9 10 7 6

1 2 3 4 5 6 9 10 7 8

1 2 3 4 5 6 7 10 9 8

1 2 3 4 5 6 7 8 9 10

* Insertion sort

3 4 5 6 7 8 9 10 1 2

1 3 4 5 6 7 8 9 10 2

1 2 3 4 5 6 7 8 9 10

* Bubble sort

3 4 5 6 7 8 9 10 1 2

Loop 1:

3 4 5 6 7 8 9 1 10 2

3 4 5 6 7 8 9 1 2 10

Loop 2:

3 4 5 6 7 8 1 9 2 10

3 4 5 6 7 8 1 2 9 10

Loop 3:

3 4 5 6 7 1 8 2 9 10

3 4 5 6 7 1 2 8 9 10

Loop 4:

3 4 5 6 1 7 2 8 9 10

3 4 5 6 1 2 7 8 9 10

Loop 5:

3 4 5 1 6 2 7 8 9 10

3 4 5 1 2 6 7 8 9 10

Loop 6:

3 4 1 5 2 6 7 8 9 10

3 4 1 2 5 6 7 8 9 10

Loop 7:

3 1 4 2 5 6 7 8 9 10

3 1 2 4 5 6 7 8 9 10

Loop 8:

1 3 2 4 5 6 7 8 9 10

1 2 3 4 5 6 7 8 9 10

* Merge sort

3 4 5 6 7 8 9 10 1 2

3 4 5 6 7 8 9 10 1 2

3 4 5 6 7 8 9 10 1 2

3 4 5 6 7 8 9 10 1 2

3 4 8 9

3 4 6 7 8 9 1 2

3 4 5 8 9 10

3 4 5 6 7 1 2 8 9 10

1 2 3 4 5 6 7 8 9 10

* Quick sort

3 4 5 6 7 8 9 10 1 2

1 2(pivot) 3 4 5 6 7 8 9 10

1 2 3 4 5 6 7 8 9 10(pivot)

1 2 3 4 5 6 7 8 9(pivot) 10

1 2 3 4 5 6 7 8(pivot) 9 10

1 2 3 4 5 6 7(pivot) 8 9 10

1 2 3 4 5 6(pivot) 7 8 9 10

1 2 3 4 5(pivot) 6 7 8 9 10

1 2 3 4(pivot) 5 6 7 8 9 10

* Heap sort
* Radix sort

10 1 2 3 4 5 6 7 8 9

1 2 3 4 5 6 7 8 9 10

1. Write a program to sort (ascending sort) the list by following methods:

* Selection sort
* Insertion sort
* Bubble sort
* Merge sort
* Quick sort
* Heap sort
* Radix sort