Algorithm 2021 Spring HW2

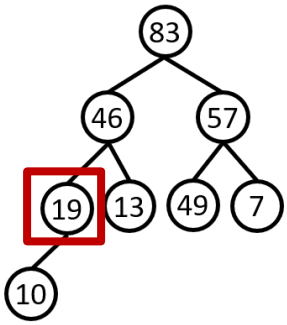
(Chapter 6 to Chapter 8)

Given 57, 19, 83, 46, 13, 49, 7, 10 in order to build a max-heap.

1. (12.5pts) Which number is the number 46's left child ?
   (a) 10
   (b) 19
   (c) 13
   (d) 49
   (e) None of above.

Ans:

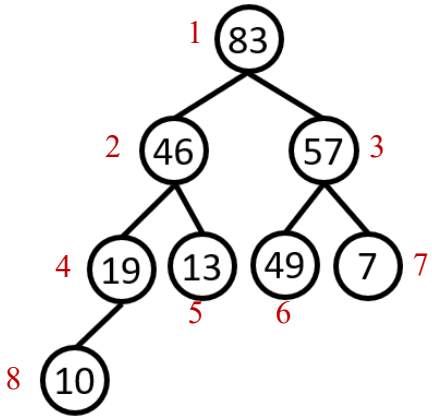
(b)



2. (12.5pts) Please choose the right array to represent the Heap in [1].
   (a) [57,46,83,19,13,49,7,10]
   (b) [57,19,83,46,13,49,7,10]
   (c) [83,46,57,19,13,49,7,10]
   (d) [83,19,57,46,13,49,7,10]
   (e) None of above.

Ans:

(c)

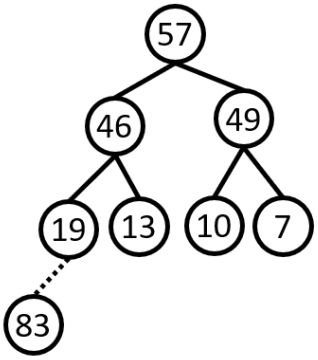


3. (12.5pts) Please choose the right array after 2 step of Heap sort in [1].
   (a) [49,46,19,10,13,7,57,83]
   (b) [49,46,10,19,7,13,57,83]
   (c) [49,46,10,19,13,7,57,83]
   (d) [49,19,46,13,10,7,57,83]
   (e) None of above.

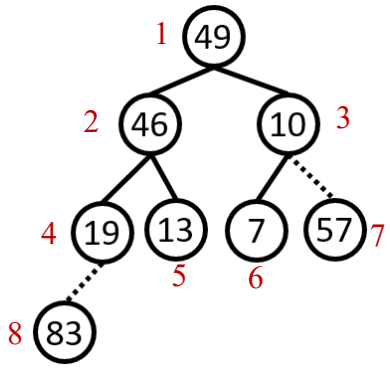
Ans:

(c)

Step1：



Step2：



Radix sort

Given 112, 375, 169, 412, 241, 842, 468, 516.

4. (12.5pts) Please choose the right result after finishing the second digit arrange.
   (a) 112, 412, 516, 241, 842, 468, 169, 375
   (b) 241, 112, 412, 842, 375, 516, 468, 169
   (c) 412, 516, 112, 241, 842, 468, 169, 375
   (d) 112, 412, 516, 241, 842, 169, 468, 375

Ans:

(a)

First round

241, 112, 412, 842, 375, 516, 468, 169

Second round

112, 412, 516, 241, 842, 468, 169, 375

For question 5, 6 and 7, please refer to the following code:

```
QUICKSORT(A, p, r)
1 if p < r then
2     q ←PARTITION(A, p, r)
3     QUICKSORT(A, p, q − 1)
4     QUICKSORT(A, q + 1, r)
```

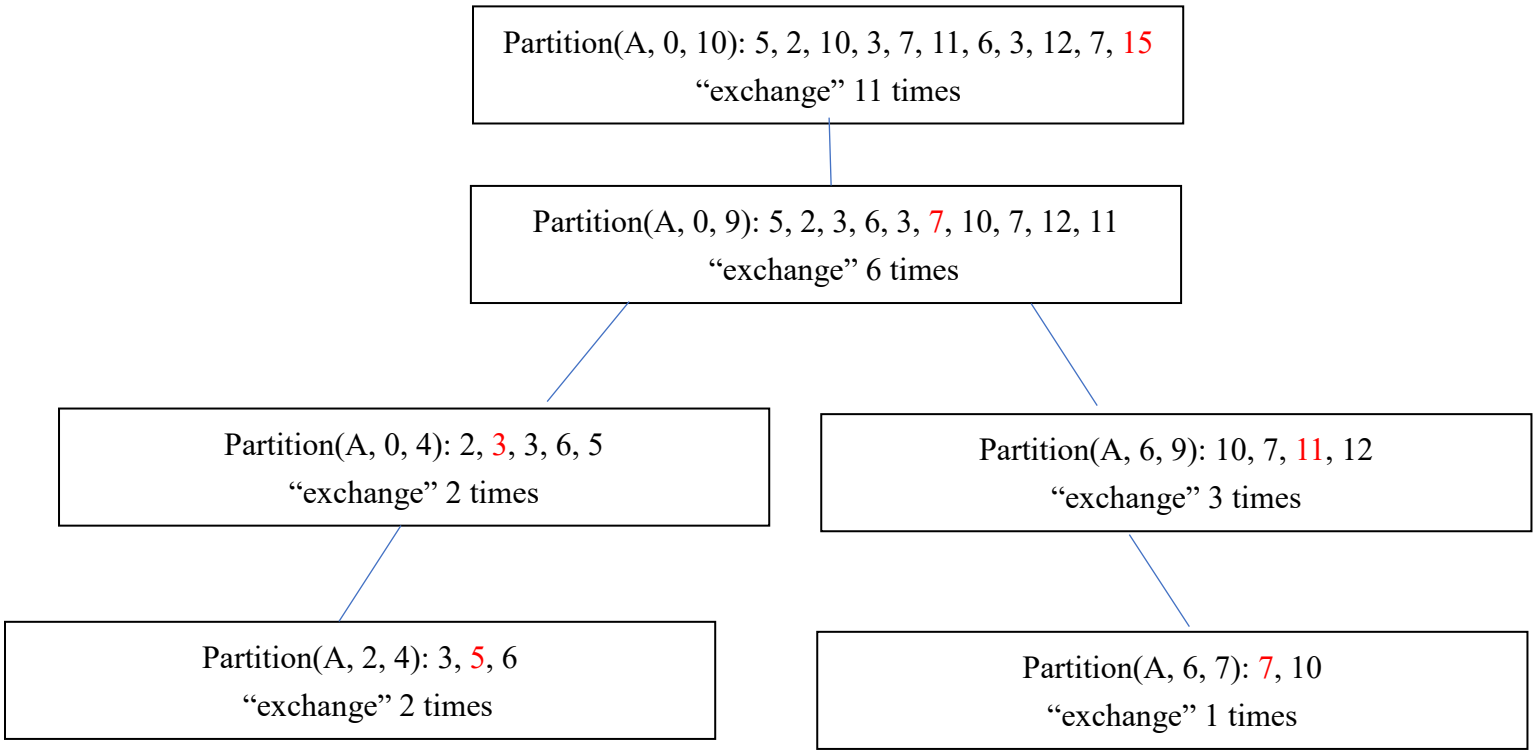
```
PARTITION(A, p, r)
1 x ← A[r]
2 i ← p − 1
3 for j = p to r − 1 do
4     if A[j] < x then
5         i ← i + 1
6         exchange A[i] ↔ A[j]
7 exchange A[i + 1] ↔ A[r]
8 return i + 1
```

5. (12.5pts) Given a list of numbers A = {5, 2, 10, 3, 7, 11, 6, 3, 12, 7, 15}, how many times are the function "exchange" being called when we call function QUICKSORT(A, 0, 10) ?

Ans:

25 times

(red number means pivot)



Therefore, 11 + 6 + 2 + 3 + 2 + 1 = 25

6. (12.5pts) Which list of numbers has the worst case of QUICKSORT?
   (a) {1, 2, 3, 4, 5, 6, 7, 8, 9}
   (b) {3, 2, 1, 4, 6, 8, 7, 5, 9}
   (c) {9, 8, 7, 6, 5, 4, 3, 2, 1}
   (d) {6, 6, 6, 6, 6, 6, 6, 6, 6}

Ans:

(a)、(c)、(d)

We need to partition each number while doing quicksort at non-increasing order or non-decreasing order sequence.

7. (12.5pts) Suppose we are implementing quicksort to sort an array. After the first partition step has been completed, the contents of the array are in the following order: 3, 5, 3, 9, 17, 14, 24, 21, 19. What is the pivot of the first partition step?

Ans:

3,9

The leftmost 3 and 9 has already partitioned. Therefore, the first pivot may be either 3 or 9.

For question 8, please refer to the following code:

```
BUCKET-SORT(A, n)
1 Input: A[1...n], where 0 ≤ A[i] < 1 for all i.
2 Auxiliary array: B[0...n − 1] of linked lists, each list initially
    empty.
3 for i ← 1 to n do
4     insert A[i] into list B[⌊n · A[i]⌋]
5 for i ← 0 to n − 1 do
6     sort list B[i] with insertion sort
7 concatenate lists B[0], B[1], ..., B[n − 1] together in order
8 return the concatenated lists
```

8. (12.5pts) Please answer the output lists(B[0],…, B[n-1]) of BUCKET-SORT on the array=(.69, .57, .99, .15, .27, .59, .61, .76, .87, .98, .01, .79, .23).

Ans:

B[0]:(.01)
B[1]:(.15)
B[2]: (.23)
B[3]:(.27)
B[4]:
B[5]:
B[6]:
B[7]: (.57,.59,.61)
B[8]: (.69)
B[9]: (.76)
B[10]: (.79)
B[11]: (.87)
B[12]: (.98,.99)

For example, insert A[1] into list B[⌊13* A[1]⌋].

⇨ Insert .69 into list B[8].

insert A[2] into list B[⌊13* A[2]⌋].

⇨ Insert .69 into list B[2].

insert A[3] into list B[⌊13* A[3]⌋].

⇨ Insert .69 into list B[3].

insert A[4] into list B[⌊13* A[4]⌋].

⇨ Insert .69 into list B[4].

insert A[5] into list B[⌊13* A[5]⌋].

⇨ Insert .69 into list B[5].

insert A[6] into list B[⌊13* A[6]⌋].

⇨ Insert .69 into list B[6].

insert A[7] into list B[⌊13* A[7]⌋].

⇨ Insert .69 into list B[7].

insert A[8] into list B[⌊13* A[8]⌋].

⇨ Insert .69 into list B[8].

insert A[9] into list B[⌊13* A[8]⌋].

⇨ Insert .69 into list B[8].

insert A[10] into list B[⌊13* A[9]⌋].

⇨ Insert .69 into list B[9].

insert A[11] into list B[⌊13* A[10]⌋].

⇨ Insert .69 into list B[10].

insert A[12] into list B[⌊13* A[11]⌋].

⇨ Insert .69 into list B[11].

insert A[13] into list B[⌊13* A[12]⌋].

⇨ Insert .69 into list B[12].