

The seal of Hanyang University is a circular emblem. It features a central shield-like shape with the Korean characters '한양' (Hanyang) inside. The words 'HANYANG UNIVERSITY' are written in a circular path around the top of the seal, and the year '1939' is at the bottom. The seal is surrounded by a decorative border of leaves and flowers.

# Lab 13: Quick Sort

---

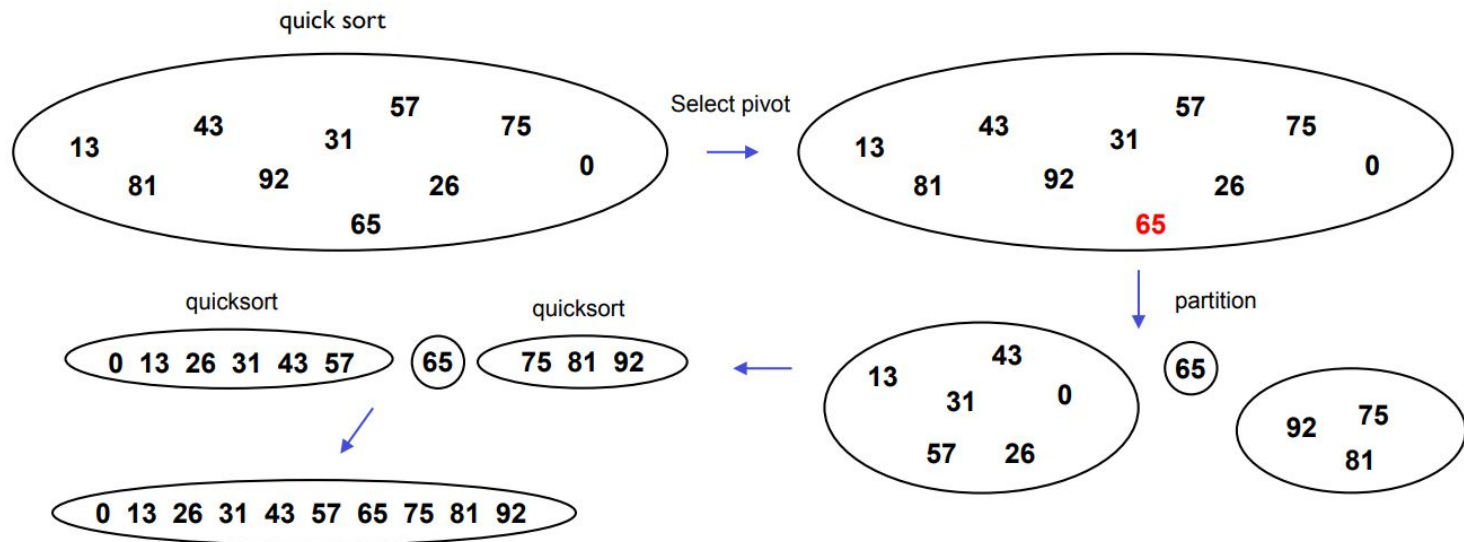
Data Structure 2023

# Quick Sort

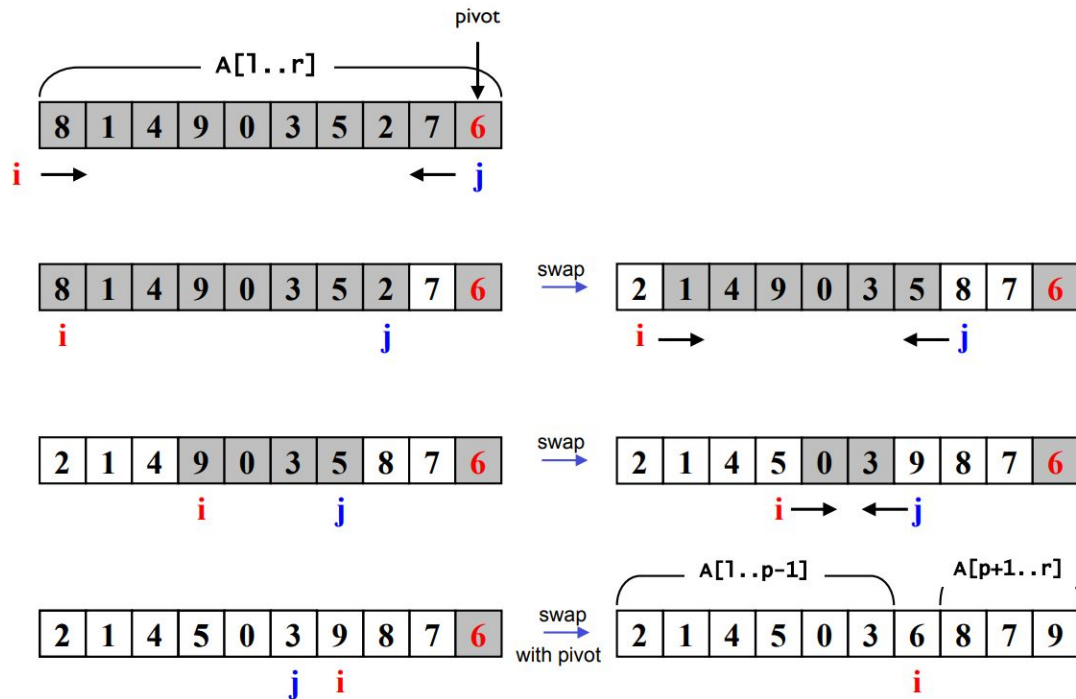
---

- Sorting
  - Sorting is putting the elements into a list in which the elements are in **increasing order**
- Quick Sort
  - Divide: partition the array  $A[l..r]$  into two subarrays  $A[l..p-1]$  and  $A[p+1..r]$ 
    - All elements in  $A[l..p-1]$  are less than or equal to a pivot element  $A[p]$
    - All elements in  $A[p+1..r]$  are greater than pivot element  $A[p]$ .
  - Conquer: sort the two subarrays  $A[l..p-1]$  and  $A[p+1..r]$  by recursive calls to quicksort.
    - Since the subarrays are sorted in place, no work is needed.

# Quick Sort



# Quick Sort



# Quick Sort ADT

---

- **Array\* CreateArray(int size)**
  - Create an array with the input size.
- **void QuickSort(Array\* array, int left, int right)**
  - Perform quick sort on the interval [left, right].
- **int Partition(Array\* array, int left, int right)**
  - Set the pivot and obtain the proper index of the pivot through the swapping.
- **void PrintArray(Array\* array, int left, int right)**
  - Print all values of the array on the interval [left, right].
- **void DeleteArray(Array\* array)**
  - Delete an array.

# Quick Sort ADT

---

## Structure

```
typedef struct Array Array;  
struct Array{  
    int size;  
    int* values;  
};
```

## Function

```
Array* CreateArray(int size);  
void QuickSort(Array* array, int left, int right);  
int Partition(Array* array, int left, int right);  
void PrintArray(Array* array, int left, int right);  
void DeleteArray(Array* array);
```

# Input & Output Example

```
Open ▾ input... Save  Open ▾ out... Save
~/Do...  ~/Do...

1 10
2 8 1 4 9 0 3 5 2 7 6

1 left : 0, right : 9, pivot : 6
2 8 1 4 9 0 3 5 2 7 6 |
3 2 1 4 9 0 3 5 8 7 6
4 2 1 4 5 0 3 9 8 7 6
5 2 1 4 5 0 3 6 8 7 9
6 left : 0, right : 5, pivot : 3
7 2 1 4 5 0 3 - - - -
8 2 1 0 5 4 3 - - - -
9 2 1 0 3 4 5 - - - -
10 left : 0, right : 2, pivot : 0
11 2 1 0 - - - - - -
12 0 1 2 - - - - - -
13 left : 1, right : 2, pivot : 2
14 - 1 2 - - - - - -
15 left : 4, right : 5, pivot : 5
16 - - - - 4 5 - - - -
17 left : 7, right : 9, pivot : 9
18 - - - - - - 8 7 9
19 left : 7, right : 8, pivot : 7
20 - - - - - - 8 7 -
21 - - - - - - 7 8 -
22
23 sorting result :
24 0 1 2 3 4 5 6 7 8 9

Tab Width: 8 ▾ Ln 2, Col 20 ▾ INS Tab Width: 8 ▾ Ln 2, Col 31 ▾ INS
```

# Assignment

---

- Due
  - ~ **2023.06.14(수) 23:59**
  - Last Commit 기준
  
- 자세한 내용은 과제 명세 PDF 파일 참고