

2019년 알고리즘

- HW 01 -

| | |
|------|-------------|
| 제출일자 | 2019.09.24. |
| 이름 | 장수훈 |
| 학번 | 201402414 |
| 분반 | 01 |

1. Insertion sort

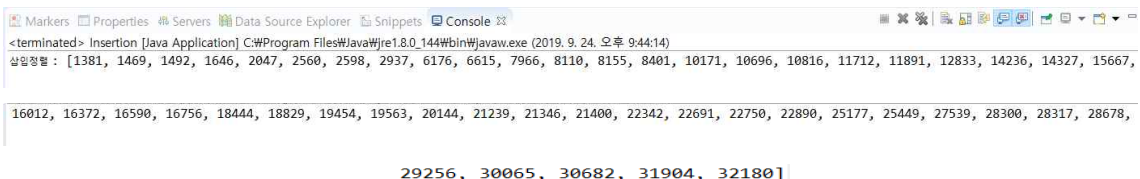
```
1 package H02;
2 import java.util.Scanner;
3
4
5 public class Insertion {
6     public static void main(String[] args) {
7         Scanner scan = new Scanner(System.in);
8         int temp;
9
10        int array[] = {1381,20144,2937,8401,31904,22750,27539,
11
12        for (int i = 1; i < array.length; i++) {
13            int mid = array[i];
14            int key = i - 1;
15
16            while (key >= 0 && mid < array[key]) {
17                array[key + 1] = array[key];
18                key--;
19            }
20            array[key + 1] = mid;
21        }
22        System.out.println("삽입정렬 : "+Arrays.toString(array));
23    }
24 }
25
```

$$T(n) = \sum_{j=2}^n \Theta(j) = \Theta(n^2)$$

코드 설명

변수 mid는 배열의 기준 key는 비교할 대상 변수이다. while문 안에서 array[key+1] array[key] 와 비교를 하여 클 경우 오른쪽으로 옮긴후 array[key+1] = mid로 저장을 해준다.

결과값



```
<terminated> Insertion [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (2019. 9. 24. 오후 9:44:14)
삽입정렬 : [1381, 1469, 1492, 1646, 2047, 2560, 2598, 2937, 6176, 6615, 7966, 8110, 8155, 8401, 10171, 10696, 10816, 11712, 11891, 12833, 14236, 14327, 15667,
16012, 16372, 16590, 16756, 18444, 18829, 19454, 19563, 20144, 21239, 21346, 21400, 22342, 22691, 22750, 22890, 25177, 25449, 27539, 28300, 28317, 28678,
29256, 30065, 30682, 31904, 32180]
```

2. Merge sort

```
1 package H02;
2
3 import java.util.Scanner;
4
5
6 public class merge {
7
8     public static void main(String[] args) {
9
10         int[] mergearr = {1381,20144,2937,8401,31904,22750,27539,6615,1492,8110,12833,118
11
12         mergesort(mergearr, 0, mergearr.length - 1);
13         System.out.println("병합정렬 : " + Arrays.toString(mergearr));
14     }
15
16     public static void mergesort(int mergearr[], int arr1, int arr2) { //분할
17         if (arr1 < arr2) {
18             int middle = (arr1 + arr2) / 2;
19             mergesort(mergearr, arr1, middle); // 앞배열 분할
20             mergesort(mergearr, middle + 1, arr2); // 뒷배열 분할
21             merge(mergearr, arr1, middle, arr2); // 앞, 뒤 배열을 가지고 병합할수로 넘어감
22         } else {
23             return;
24         }
25     }
26
27     public static void merge(int mergearr[], int arr1, int middle, int arr2) { //합병
28         int i = arr1, // 첫번째 시작위치
29             j = middle + 1, // 두번째 시작위치
30             k = arr1;
31         int result[] = new int[mergearr.length];
32         while (i <= middle && j <= arr2) {
33             if (mergearr[i] <= mergearr[j]) {
34                 result[k++] = mergearr[i++];
35             } else {
36                 result[k++] = mergearr[j++];
37             }
38         }
39         while (i <= middle)
40             result[k++] = mergearr[i++];
41         while (j <= arr2)
42             result[k++] = mergearr[j++];
43         for (i = arr1; i <= arr2; i++)
44             mergearr[i] = result[i];
45     }
46 }
47
```

$$T(n) = \Theta(n \lg n)$$

코드 설명

우선 배열을 나누기 위해서 mergesort 함수로 배열을 가져간다. mergesort 함수에선 배열을 분할하는 작업을하는데 19번째 line은 앞배열 분할 20번째 line은 뒷배열 분할이다. 이후 병합할 때 merge 함수로 앞에 나눈 배열들은 인자로 들어간다.

merge함수에서는 l는 첫 번째 시작위치 j는 두 번째 시작위치, k 는 배열에 정렬된 원소를 저장할 위치를 설정하기위한 변수이다. merge한 배열들은 최종배열인 result배열에 넣어준 후 메인 함수에서 Array.toString으로 출력하였다.

결과값

```
<terminated> merge [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (2019. 9. 24. 오후 9:55:38)
병합정렬 : [1381, 1469, 1492, 1646, 2047, 2560, 2598, 2937, 6176, 6615, 7966, 8110, 8155, 8401, 10171, 10696, 10816, 11712, 11891, 12833, 14236, 14327, 15667, 16012, 16372, 16590, 16756, 18444, 18829, 19454, 19563, 20144, 21239, 21346, 21400, 22342, 22691, 22750, 22890, 25177, 25449, 27539, 28300, 28317, 28678, 29256, 30065, 30682, 31904, 32180]
```