

REPORT



제 목 : 알고리즘 과제#2

과 목 명 : 알고리즘

담당교수 : 우 진 운 교수님

이 름 : 조 정 민

학 번 : 32164420

제 출 일 : 2020. 04. 24



단국대학교
Dankook University

소스 코드(MergeSort)

```
public static void main(String[] args) {
    int size[] = { 1000, 5000, 10000, 20000, 50000, 100000 };
    int[][] store = new int[6][]; // 난수를 저장할 배열

    for (int i = 0; i < 6; i++)
        store[i] = new int[size[i]+10]; // 배열에 대한 size

    System.out.println("MergeSort의 결과 : ");
    System.out.println(" [1000] [5000] [10000] [20000] [50000] [100000]");
    System.out.print(" ");
    for (int i = 0; i < 10; i++) { // 10개의 데이터를 테스트하기 위한 for loop
        for (int j = 0; j < 6; j++) { // 각각의 크기만큼 6번 수행
            for (int k = 0; k < size[j]; k++) // 할당받은 크기만큼
                store[j][k] = (int) (Math.random()*size[j]); // 난수 입력
        }
        for (int l = 0; l < 6; l++) {
            MergeSortClass merge = new MergeSortClass(store[l], size[l]);
            long start = System.nanoTime(); // 합병정렬 시작시간
            store[l] = merge.MergeSortCall();
            long end = System.nanoTime(); // 합병정렬 끝난 시간
            long execute=end-start; // 합병정렬에 걸린 시간
            System.out.print(execute + " ");
        }
        System.out.println();
        System.out.print(" ");
    }
    System.out.println("");
} // 소프트웨어학과 32164420 조정민
```

실행 결과

MergeSort의 결과 :

[1000]	[5000]	[10000]	[20000]	[50000]	[100000]
531600	1082700	2285100	4921300	7511600	11792900
84800	449000	955600	2063300	5559500	11563800
88600	441100	958100	1936000	5328100	11412200
72400	427200	935200	1951600	5637700	11165400
73200	445800	942000	1978800	5359800	11485600
72400	446800	929600	1981000	6115600	11349900
96300	472400	916700	1888600	5235900	11064400
72000	446300	902900	1904400	5245600	10985900
72600	517100	946700	2136500	5108100	11384600
74500	426100	942900	1899400	5082700	11256900

소스 코드(QuickSort)

```
public static void main(String[] args) {
    int size[] = { 1000, 5000, 10000, 20000, 50000, 100000 };
    int[][] store = new int[6][]; // 난수를 저장할 배열

    for (int i = 0; i < 6; i++)
        store[i] = new int[size[i]+10]; // 배열에 대한 size

    System.out.println("QuickSort의 결과 : ");
    System.out.println(" [1000] [5000] [10000] [20000] [50000] [100000]");
    System.out.print(" ");
    for (int i = 0; i < 10; i++) { // 10개의 데이터를 테스트하기 위한 for loop
        for (int j = 0; j < 6; j++) { // 각각의 크기만큼 6번 수행
            for (int k = 0; k < size[j]; k++) // 할당받은 크기만큼
                store[j][k] = (int) (Math.random()*size[j]); // 난수 입력
        }
        for (int l= 0; l < 6; l++) {
            QuickSortClass quick = new QuickSortClass(store[l], size[l]);
            long start = System.nanoTime(); // 측정할 시작시간
            store[l] = quick.QuickSortCall();
            long end = System.nanoTime(); // 측정할 끝난 시간
            long execute=end-start; // 측정할에 걸린 시간
            System.out.print(execute + " ");
        }
        System.out.println();
        System.out.print(" ");
    }
    System.out.println("");
} // 소프트웨어학과 32164420 조정민
```

실행 결과

QuickSort의 결과 :

[1000]	[5000]	[10000]	[20000]	[50000]	[100000]
262600	733000	1400700	2669900	3452000	7388700
56200	298300	610300	1327100	3582500	7535700
69700	284500	634100	1364600	3523900	7532300
47400	298500	633500	1257900	3441100	7249400
45500	293800	622400	1308700	3388000	7797000
46600	296300	622600	1285700	3724800	7818200
47400	272400	575300	1231400	3474300	7147500
45300	315300	626500	1263100	3401400	7758500
47100	300900	636200	1237200	3466600	7553000
47100	273300	577100	1729300	3381700	7135200

합병	1	2	3	4	5	6	7	8	9	10
1000	531600	84800	88600	72400	73200	72400	96300	72000	72600	74500
5000	1082700	449000	441100	427200	445800	446800	472400	446300	517100	426100
10000	2285100	955600	958100	935200	942000	929600	916700	902900	946700	942900
20000	4921300	2063300	1936000	1951600	1978800	1981000	1888600	1904400	2136500	1899400
50000	7511600	5559500	5328100	5637700	5359800	6115600	5235900	5245600	5108100	5082700
100000	11792900	11563800	11412200	11165400	11485600	11349900	11064400	10985900	11384600	11256900

क्व	1	2	3	4	5	6	7	8	9	10
1000	262600	56200	69700	47400	45500	46600	47400	45300	47100	47100
5000	733000	298300	284500	298500	293800	296300	272400	315300	300900	273300
10000	1400700	610300	634100	633500	622400	622600	575300	626500	636200	577100
20000	2669900	1327100	1364600	1257900	1308700	1285700	1231400	1263100	1237200	1729300
50000	3452000	3582500	3523900	3441100	3388000	3724800	3474300	3401400	3466600	3381700
100000	7388700	7535700	7532300	7249400	7797000	7818200	7147500	7758500	7553000	7135200

평균	1000	5000	10000	20000	50000	100000
합병 정렬	123840	515450	1071480	2266090	5618460	11346160
क्व 정렬	71490	336630	693870	1467490	3483630	7491550