

프로그래밍 과제 03

1. 다양한 정렬 알고리즘들을 다양한 크기의 테스트 데이터에 대해서 실행시간을 분석하는 프로그램을 작성한다. 분석할 정렬 알고리즘은 다음과 같이 9가지이다.

- (1) 버블 정렬(bubble sort)
- (2) 선택정렬(selection sort)
- (3) 삽입 정렬(insertion sort)
- (4) 합병정렬(merge sort)
- (5) 빠른정렬(quicksort) - 마지막 값을 피벗으로 선택
- (6) 빠른정렬(quicksort) - 첫번째 값, 가운데 위치의 값, 그리고 마지막 값 중에서 중간값을 pivot으로 선택
- (7) 빠른정렬(quicksort) - pivot을 랜덤하게 선택
- (8) 힙정렬(heap sort)
- (9) C, C++ 혹은 Java등 자신이 사용하는 언어의 표준 라이브러리가 제공하는 정렬 알고리즘

다음과 같은 테스트 데이터를 사용하여 알고리즘의 실행시간을 분석한다. 테스트 데이터의 크기 N은 1,000, 10,000, 100,000인 3가지 경우를 테스트한다. 각각의 크기 대해서 다시 다음과 같은 2종류의 데이터를 생성하여 사용한다.

- 랜덤 데이터: 1에서 N 사이의 N개의 정수들을 랜덤하게 생성한 데이터 (중복된 값이 있을 수 있음)
- 거꾸로 정렬된 데이터: 1에서 N까지 N개의 정수들이 거꾸로 정렬된 데이터 (예를 들어 N=1000이면 1000, 999, 998, 997, ..., 2, 1)

즉, 총 6가지의 테스트 데이터를 사용한다. 랜덤 데이터에 대해서 10개의 테스트 데이터를 생성하여 실행한 후 실행시간의 평균을 구한다. 알고리즘의 실행시간은 입출력 시간 등을 제외하고 순수하게 정렬 알고리즘이 실행되는 시간만을 측정해야 한다.

출력형식:

다음과 같은 형태의 테이블을 출력한다. 테이블의 세로 축은 8가지 정렬 알고리즘을 나타내고 가로 축은 6가지 유형의 테스트 데이터를 나타낸다. 각각에 대해서 알고리즘의 (평균) 실행시간을 초 단위로 출력한다. (테이블에 써진 값은 가짜 값이므로 신경쓰지 말것). **아래 테이블에서 몇몇 경우들을 실행시간이 10분을 초과하거나 혹은 stack overflow 오류가 날 것이다. 그런 경우에 대해서는 실행을 포기해도 된다. 즉, 그런 경우들은 제외하고 실행한다.**

	Random1000	Reverse1000	Random10000	Reverse10000	Random100000	Reverse100000
Bubble	0.001	0.001	0.01	0.01	1.235	1.235
Selection						
Insertion						
Merge						
Quick1				...		
Quick2						
Quick3						
Heap						
Library						