

알고리즘 HW1

컴퓨터공학부 최덕경

2016-10399

Selection 알고리즘이란 주어진 배열에서 k번째로 작은 값을 찾아내는 알고리즘을 말한다. 이번 과제에서는 Randomized-select, Deterministic-select 알고리즘을 직접 구현해보므로써 그 동작 원리를 제대로 이해하고, 직접 checker 프로그램을 만들어서 구현한 알고리즘이 올바르게 동작하는지 검사해볼 것이다. 그리고 직접 동일한 데이터에 대하여 두 알고리즘에 소요되는 시간을 측정하여 비교해보고, asymptotic 시간 복잡도를 각각 계산하여 숨어 있는 상수의 비율을 파악해볼 것이다. 이를 통해 최종적으로 어떤 알고리즘의 성능이 더 좋은지 판단하게 될 것이다.

- 사용 언어 : Python 3.6.7
- 실험 환경 : Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-45-generic x86_64)
- 실험 방법 : python3 test.py
- **checker(a, n, k, ans)** 함수의 동작 방식 및 시간복잡도
길이가 n인 배열 a에서 k번째($0 \leq k \leq n-1$)로 작은 원소가 ans라면, 배열 a에서 ans보다 작은 값은 k개만큼 있어야 한다. 그렇다면 True를 반환하고, 아니라면 False를 반환한다. 또한 이 함수는 단순히 배열 a를 선형 탐색하며 ans보다 작은 값의 개수를 세기 때문에, 시간복잡도는 $O(n)$ 이다.
- **Experiment** : test.py 파일에는 함수의 소요 시간을 측정하는 runtime 함수가 구현되어 있고, 이를 이용하여 실험을 위해 제공된 네 개의 입력 파일 각각에 대해 두 알고리즘의 소요 시간을 측정하고 알고리즘의 결과값이 맞는지 검사하는 코드가 들어 있다. 단 Randomized-select 알고리즘의 경우 실행할 때마다 피벗을 어떻게 잡느냐에 따라 실행 시간이 달라질 수 있으므로 5번 실행하고 평균값을 계산하도록 하였다. (test.py 모듈 내부에 정의된 runtime 함수와 hw1.py 모듈에 정의된 checker 함수를 활용)

```
===== 1th test case =====
<Randomized Select Algorithm>
1. run time : 0.014817704912275076
2. result : 411681976
3. correctness : True
<Deterministic Select Algorithm>
1. run time : 0.0443045231513679
2. result : 411681976
3. correctness : True
<Run time ratio = Randomized/Deterministic>
0.3344512898073609

===== 2th test case =====
<Randomized Select Algorithm>
1. run time : 0.03151836059987545
2. result : 273794050
3. correctness : True
<Deterministic Select Algorithm>
1. run time : 0.10755744483321905
2. result : 273794050
3. correctness : True
<Run time ratio = Randomized/Deterministic>
0.29303746150485926

===== 3th test case =====
<Randomized Select Algorithm>
1. run time : 0.09991697249934077
2. result : 12029152
3. correctness : True
<Deterministic Select Algorithm>
1. run time : 0.3669148338958621
2. result : 12029152
3. correctness : True
<Run time ratio = Randomized/Deterministic>
0.27231652489607233

===== 4th test case =====
<Randomized Select Algorithm>
1. run time : 0.37152938209474085
2. result : 543945926
3. correctness : True
<Deterministic Select Algorithm>
1. run time : 1.1951016900129616
2. result : 543945926
3. correctness : True
<Run time ratio = Randomized/Deterministic>
0.31087679416695607
```

▶ 네 개의 입력 파일 각각에 대한 두 알고리즘의 [1. 실행 시간 2. 출력값 3. Correctness]
(ith test case는 i.txt 파일을 나타낸다.)

- **Ration 비교** : 위 실험에서 측정된 두 알고리즘의 소요 시간과 그 비율을 정리하면 다음과 같다.

입력 파일 (데이터 크기)	(1) Randomized-select	(2) Deterministic-select	Ratio (1)/(2)
1.txt (n = 20,000)	0.014817704912275076 초	0.0443045231513679 초	약 0.33
2.txt (n = 60,000)	0.03151836059987545 초	0.10755744483321905 초	약 0.29
3.txt (n = 200,000)	0.09991697249934077 초	0.3669148338958621 초	약 0.27
4.txt (n = 600,000)	0.37152938209474085 초	1.1951016900129616 초	약 0.31

두 알고리즘의 시간 복잡도는 동일하게 $O(n)$ 이지만 실행 시간의 비가 대략 0.3 정도를 유지하는 것으로 보아 [Randomized-select 시간 복잡도의 상수 \approx (0.3) X Deterministic-select 시간 복잡도의 상수]임을 유추할 수 있다.

- 마지막으로, 다음 그래프를 통해 구현한 두 알고리즘이 $O(n)$ 의 시간 복잡도 양상을 가짐을 알 수 있다.
(둘 다 거의 일직선(선형)을 이루면서 단조 증가하고 있기 때문이다.)

