

알고리즘 분석 과제3

컴퓨터공학과 2019110630 박규리

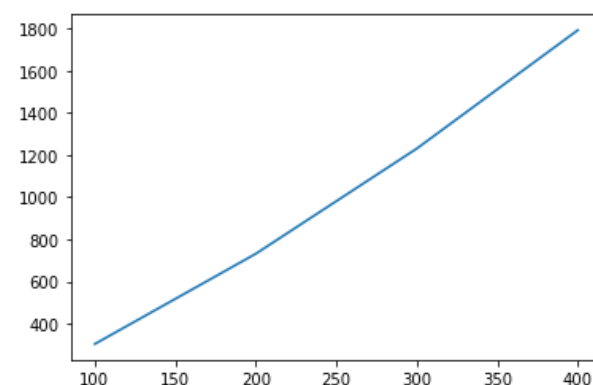
(1) Quick Sort

Random 라이브러리를 import하여 random_List 함수를 정의하여 1부터 n까지의 랜덤 리스트를 생성한다. Pivotpoint는 pivotitem의 index로 pivotitem의 위치를 나타낸다. quicksort 함수와 partition 함수를 정의하고 avg_comparison 함수를 만들었다. Avg_comparison 함수는 평균 데이터 비교횟수를 구하는 함수로 밑에 그래프를 그리기 위해 정의하였다.

Matplotlib.pyplot을 import하여 최종적으로 가로축에 n, 세로축에 평균 데이터 비교 횟수를 표시하는 그래프를 생성하였다. 이 때 n = 100 일 때 289.53, n = 200 일 때 739.57, n = 300 일 때 1209.71, n = 400 일 때 1705.5가 나오고 그 점들을 이어주어 그래프가 나타나게 된다. 점들을 이어주는 선이 의미하는 바는 **시간 복잡도**이고 그래프 모양으로 유추해 보았을 때 $O(n^2)$ 과 $O(n\log n)$ 사이의 그래프 형태이다.

```
import matplotlib.pyplot as plt

plt.plot([100, 200, 300, 400], [avg_comparison(100), avg_comparison(200), avg_comparison(300), avg_comparison(400)])
plt.show()
print(avg_comparison(100))
print(avg_comparison(200))
print(avg_comparison(300))
print(avg_comparison(400))
```



289.53
739.57
1209.71
1705.5

(2) 큰 정수 곱셈 알고리즘

두 정수 a와 b의 곱셈을 실행하는 함수 prod2를 정의했다. 먼저 a의 자릿수를 n1, b의 자릿수를 n2라고 하고 n1과 n2중 더 큰 수를 n이라고 한다.

if : a와 b 둘 중 하나가 0이면 곱한 값 역시 0이 된다.

elif : 만약 n이 threshold보다 작다면 $a*b$ 값을 리턴한다. 이때, threshold=2로 설정했다.

else :

m은 n을 2로 나눈 몫이다. x는 a를 10^m 으로 나눈 몫이고, y는 나머지이다. w는 b를 10^m 으로 나눈 몫이고 z는 나머지이다. 이에 따라 r,p,q를 구한 후

$p*10^{2m}+(r-p-q)*10^m + q$ 값을 리턴하면 a와 b의 곱셈 연산의 결과가 나오게 된다.

실제로 디버깅 해보면 prod2(a,b)의 값과 $a*b$ 의 값이 동일하게 출력되는 것을 알 수 있다.