**과제1 보고서**

**컴퓨터공학과 2019102144 공태현**

(1) => **소스코드로 알고리즘 구현**

(2) 다음의 문제 크기 n에 대해 알고리즘 A, B가 종료될 때까지의 시간을 측정하여 다음 테 이블에 작성하라.

|  |  |  |
| --- | --- | --- |
| n | 알고리즘 A | 알고리즘 B |
| 5,000 | 1.0762500762939453 | 0.007346153259277344 |
| 10,000 | 4.229114294052124 | 0.01674818992614746 |
| 15,000 | 9.709459066390991 | 0.028728961944580078 |
| 20,000 | 17.72713303565979 | 0.040712833404541016 |
| 30,000 | 38.40761709213257 | 0.06674003601074219 |
| 40,000 | 69.21380186080933 | 0.09817790985107422 |
| 80,000 | page1image47875728 | 0.2725999355316162 |

설정 : 동일한 원소를 갖고 있는 배열 2개를 생성하여 각각 알고리즘 A, B 에 넣어주어 시간을 계산하였다. 또한 n 의 값이 변화할 때마다 새로운 배열 2개를 위와 같이 생성하였다.

(3) 데이터의 크기가 2배, 3배, 4배 될 때 수행시간의 비율을 구하는 것

|  |  |  |
| --- | --- | --- |
| n’/n | A | B |
| 2 | (3.92 + 4.19 + 3.96 + 3.9)/4 = **3.99** (평균값) | (2.29 + 2.44 + 2.32 + 2.41 + 2.78)/5 = **2.45** (평균값) |
| 3 | (8.99 + 9.08)/2 = **9.04** (평균값) | (3.93 + 3.99)/2 = **3.96** (평균값) |
| 4 | (15.95 + 16.36 )/2 = **16.16** (평균값) | (5.58 + 5.88 + 6.7) / 3 = **6.05** (평균값) |

설정 : A는 (2) 의 테이블 값의 소수 3번째 자리에서 반올림하여 계산하고, B는 테이블 값의 소수 4번째자리까지만 사용하여 계산하였다.

(4) (3)의 결과에서 관찰한 내용과 n=40,000일 때의 결과를 이용하여 n=50,000,000 일 때의 알고리즘 A의 수행시간을 추정한다. 추정 결과를 year 단위로 표시하라. 추정 방법에 대해 설명한다.

추정 과정) n = 40,000 일 때, 걸린 시간은 약 ‘69초’ 이며 (3) 의 결과로 보았을 때 n’/n 의 값이 제곱의 형태로 증가하고 있으며 이는 시간 복잡도 O(n^2) 과 맞는 형태를 보여주고 있다. 따라서, n’ / n = 50,000,000 / 40,000 = 1250 -> 69초 \* 1250 \* 1250 의 시간이 걸릴 것이라는 것을 예측할 수 있다. 이를 year 단위로 표현하면, ( 69 / 60 / 60 / 24 / 365) year \* 1250 \* 1250 = **‘3.42년’** 이다.

(5) 그래프를 그리고 a 값을 추정하라.

차트이(가) 표시된 사진

자동 생성된 설명

**추정 과정**

먼저, (2)의 테이블을 보고 해당하는 점을 찍고, n = 30000, 40000 에 해당하는 점을 이은 직선과

n = 5000, 10000 을 이은 직선의 기울기를 비교해보았다. 이를 통해 해당 그래프는 기울기가 조금씩 증가하고 있는 그래프라는 것을 알 수 있었고 이는 nlnn 을 미분한 lnn + 1 의 모습과 비슷하다는 것을 알 수 있었다. 즉, O(nlgn) 의 모습과 맞는 형태를 띈다는 것을 알 수 있다. 또한 n = 5000 의 값을 대입하여 a \* 5000 \* log2 (5000) 과 5000 \* log2 (5000) 의 값을 비교하여 a 의 값을 추정할 수 있다. 이를 n = 5000, 10,000 . . . 80,000 까지 적용하여 a 의 값의 평균 값을 유추하였다.

**계산 과정**

n = 5000 -> log 값 : 12.29 -> nlog[2]{n} 값 : 5000 \* 12.29

-> a 값 : anlog[2]{n} / nlog[2]{n} = 0.0073 / (5000 \* 12.29) = **1.19e-7**

n = 10000 -> log 값 : 13.29 -> nlog[2]{n} 값 : 10000 \* 12.29

-> a 값 : anlog[2]{n} / nlog[2]{n} = 0.0167 / (10000 \* 12.29) = **1.36e-7**

n = 15000 -> log 값 : 13.87

-> a 값 : **1.38e-7**

n = 20000 -> log 값 : 14.29

-> a 값 : **1.42e-7**

n = 30000 -> log 값 : 14.87

-> a 값 : **1.5e-7**

n = 40000 -> log 값 : 15.29

-> a 값 : **1.61e-7**

n = 80000 -> log 값 : 16.29

-> a 값 : **2.09e-7**

**a 값의 평균 : 1.51e-7**

**따라서 해당 그래프는 대략 1.51e-7log[2]{n} 의 모습을 띌 것으로 추측할 수 있다.**

(6) (5)의 결과를 이용하여 알고리즘 B를 컴퓨터로 1분간 수행할 때 해결할 수 있는 문제의 크기 n' 를 추정하라. 추정 방법을 설명한다.

추정 방법 : (5) 의 그래프를 이용하여 계산한다. Y 축이 걸린 시간(sec)이므로, y가 60일 때의 n의 값을 추정하면 될 것이다. 따라서 1.51e-7log[2]{n} = 60 일 때의 n 값을 구한다.

log[2]{n} = 60/1.51 \* 10^7

**대략, n = 2^(60/1.51\*10^7) 으로 추정할 수 있다.**