|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3 주차 문제해결기법 보고서 | | | | | |
| 문제 | 조직개편 | | | | |
| 분반 | 4 | 학번 | 12141556 | 이름 | 신도빈 |
| 알고리즘 개요 | 시간제한이 1초이기 때문에 최소 공통 부모를 찾는 함수를 최적화하지 않으면 timelimit이 걸린다. LCA 함수를 먼저 살펴보겠다. Lca[a]라는 배열에 K로 초기화 해준다. loop안에서 a= parent[a]함으로써 a를 a의 부모로 바꿔주고 이를 반복해서 a의 모든 조상들이 lca배열에 k를 갖게 된다. 그다음 loop에선 b의 lca[b]값이 k인 것을 찾는다 만약 찾았다면 b가 최소 공통 부모일 것이고 아니라면 b=parent[b]해줌으로써 계속 올라가며 찾는다. 만약 끝까지 찾아도 없다면 이들의 공통 부모는 root이므로 0을 return 해준다.  이제는 document under 그리고 kinds를 살펴 보자. Under가 가장 간단하니 먼저 다루겠다. Under는 단순하게 a의 부모를 b로 선언해주면 된다. 따라서 parent[a]=b; 해주면 된다. 다음으로 document를 살펴보자 getlca로 최소 공통 조상을 찾고 loop(a가 최소 조상이 아닐 때까지) 안에서 a의 document를 c로 선언 a=parent[a]하면 a부터 최소 조상까지 모든 document 가 c로 초기화 될 수 있다. b또한 다른 loop에서 a처럼 해준다. 마지막으로 kinds를 살펴보겠다. Set 집합을 사용했다. a와 b의 공통 조상을 구하고 document에서 처럼 loop(a가 공통조상이 아닐 때까지) 집합에 document type을 insert한다 set은 중복되지 않는 정보를 입력 받는다. 따라서 b까지 위의 방법대로 하고 set의 size를 print하면 a와b사이의 사용되는 document 종류의 수가 나온다. | | | | |
| 성능분석 | Document와 getlca의 최악의 경우는 공통부모가 root일 때 일 것이다. Tree의 height만큼의 시간이 걸릴 것이다. Tree가 balanced 되있다면 h=logN이겠지만 최악의 경우 h=N일 수 있으니 시간 복잡도는 O(N)이다. 공간 복잡도는 사원의 수만큼이니 O(N)이다. | | | | |
| 실험결과 | 사용 시간 0.9초 사용 메모리는 393216byte이다. 제한조건인 1초에 아슬아슬하게 만족했다. 공통된 부모를 찾을 때 a와b중 높이가 더 높은 곳에서부터 부모를 찾으면 시간을 더 단축 할 수 있을 것 같다. | | | | |
| 비고 |  | | | | |