|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1주차 - A 문제해결기법 보고서 | | | | | |
| 문제 | 아르바이트 - 1 | | | | |
| 분반 | 002 | 학번 | 12141540 | 이름 | 박영창 |
| 알고리즘 개요 | * Testcase의 개수 T를 입력 받는다. (1<=T<=500000) * 물건의 가격 p, 동전의 종류 n을 차례대로 입력 받는다. (1<=p<=10000, 1<=n<=14) * 동전의 액면가 배열인 coin[14]에 coin[0] 부터 coin[n]까지 입력 받는다. * 손님이 10000원을 지불했을 때 거스름돈의 액수를 맞춰야 하므로, 물건의 가격 p를 이용하여 거스름돈을 구한다. => (p = 10000 – p) * 입력되는 동전의 액면가가 오름차순인 것과 액면가가 작은 동전이 큰 동전의 약수라는 점을 이용하면, 가장 작은 액면가의 동전이 거스름돈으로 나누어 떨어진다면 입력 받은 동전들로 거스름돈을 만들 수 있다는 것을 알 수 있다. 만들 수 없는 경우에는 조건문에 진입하지 않고 바로 동전의 개수를 나타내는 cnt값을 출력한다.(cnt값은 처음에 0으로 초기화 시킨다.) * 위의 조건으로 (p % coin[0] == 0) 일 때, p가 가장 액면가가 큰 동전보다 큰 가격이라면, while 반복문을 사용해p가 0이 될 때까지 p에서 그 동전의 액면가를 빼 주고, cnt값을 1 올려준다. 만약 p가 동전의 액면가보다 작다면 while문을 break하고, 그 다음 동전에 대해 같은 작업을 반복한다. * 연산을 끝낸 뒤에, cnt값을 출력한 다음 cnt를 다시 0으로 설정한다. 그 다음, T를 입력 받는 것을 제외한 위 과정을 T-1번 반복한다. | | | | |
| 성능분석 | 물건의 가격과 동전의 액면가에 따라 소요시간이 바뀐다.  동전이 나누어 떨어진다면 O(2n), 나누어 떨어지지 않는다면 O(n) | | | | |
| 실험결과 | 소요시간 : 1.08s  사용된 메모리 공간 : 131.072KB | | | | |
| 비고 |  | | | | |