

1. 충돌 횟수의 총합

[합성수]

 $M_1(7000)$: 2305번 충돌 $M_3(12000)$: 2733번 충돌 $M_5(22000)$: 1359번 충돌

[소수]

 $M_2(6997)$: 1821번 충돌 $M_4(11117)$: 1235번 충돌 $M_6(22307)$: 655번 충돌

Hash Prime의 값이 커지면 충돌 횟수가 적어지는 경향을 보였다. 합성수의 경우 M_1, M_3, M_5 의 값은 증가했지만 충돌 횟수는 한번 증가 했다가 감소하는 추세를 보였다. 항상 감소하는 것처럼 보이지는 않았다. 그에 비해 소수의 경우 M_2, M_4, M_6 의 값이 증가함에 따라 충돌 횟수는 계속 감소하는 추세를 보였다.

그리고 합성수와 소수의 크기를 비슷한 값들로 구성을 하여 충돌 횟수를 비교해봤는데, 소수로 Hash Prime을 구성했을 때 거의 항상 충돌이 더 적은 것을 알 수 있었다. 값이 커짐에 따라 충돌 값은 2배 가까이 차이가 났다.

이로써 해시 값을 Division Method로 구할 때 Modifier값을 소수로 구하는 이유를 알 수 있었다.

2. 비교 횟수의 총합

hw6와 같이 hash에 값이 1개만 있을 때도 key comparison을 1로 센다면, 7216번 비교하게 되었다.

빈 hash에 값을 1개를 넣었을 때 key comparison을 0으로 센다면, 1459번 비교하게 되었다.

둘의 차이는 굉장히 크다. 기본적으로 등차수열의 시작이나 끝을 어디로 할까 에서 그 차이가 발생하게 되는데, 등차수열의 합 공식만 생각을 해봐도 그 차이가 꽤 커지게 된다.

c_i 부터 등차를 -1 로 하여 c_i 부터 1까지 등차의 합을 구하게 되면 그 값은 $c_i * (c_i + 1) / 2$ 이고, $c_i - 1$ 부터 1까지 등차의 합을 구하게 되면 그 값은 $(c_i - 1) * c_i / 2$ 가 된다. 전자의 경우로 하게 되면 5757개의 요소들이 각각 기본적으로 한 번씩 비교하게 되므로 비교 횟수는 5757보다 무조건 크게 나온다. 하지만 후자의 경우로 하게 되면 5757개의 요소들 중 해시 충돌이 일어나지 않은 값들은 비교 횟수에서 제외되므로 5757보다 더 작은 숫자가 나오게 된다.