

# Matrix Multiplication 2K

Consider the following C++ code, it constructs  $n \times n$  matrices A,B and vector V from matrix  $C=A \times B$  which you need to calculate.

```
uint32_t n, i, j, d1, p1, r1, m1, d2, p2, m2, A[n][n], B[n][n];
uint64_t C[n][n], V[n];
//here you need to read n, p1, d1, r1, m1, p2, d2, r2, m2 from input.
for (i=0; i<n; ++i)
    for (j=0; j<n; ++j) {
        d1 = d1 * p1 + r1;
        d2 = d2 * p2 + r2;
        A[i][j] = d1 >> (32 - m1);
        B[i][j] = d2 >> (32 - m2);
    }
//here you need to calculate C=A*B
for (i=0; i<n; ++i) {
    V[i] = 0;
    for (j=0; j<n; ++j)
        V[i] ^= C[i][j];
}
//here you need to output V[0], V[1], ..., V[n-1], separated by spaces.
```

## Input

You are given integers  $n, p1, d1, r1, m1, p2, d2, r2, m2$ , separated by spaces.

$1 \leq n \leq 2048, 1 \leq m1, m2 \leq 26, 0 \leq p1, d1, r1, p2, d2, r2 < 2^{32}$ .

## Output

You need to output  $V[0], \dots, V[n-1]$ , separated by spaces.

## Example

**Input:**

4 1664525 0 1013904223 26 1664525 1 1013904223 26

**Output:**

3929555766216722 770418013451752 7738542081270672 4286515685761206

## P.S.

There are 4 tests with  $n \approx 2048$ . My best solution time — ~3.2s (~0.8s per test) which is 20 times faster than time limit.

Also try the challenge [MATRMUL](#), to get AC there your time here should be less than 14s for cubic algorithm and 16s for Strassen's.