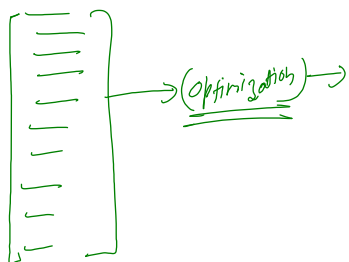
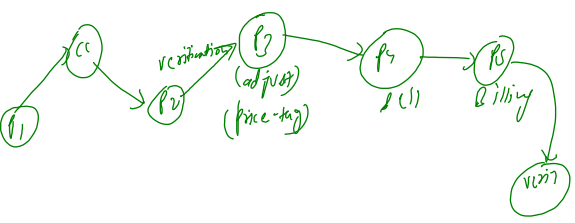


→ [ (6-2) month L2 (9-12)  
 [ (7-9) month (M W F)  
 → have → recursion (L1, Recursion) → Test  
Learn: Java

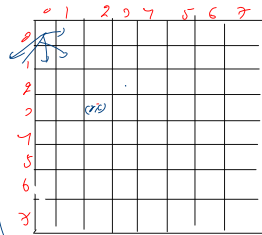
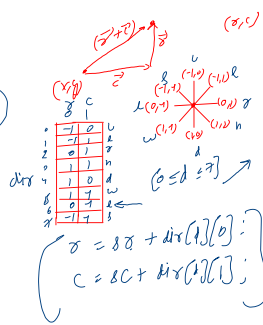


- ① Rajneesh Kumar
- ② 2020 → NSIT CNSUD
- ③ Mohit Sir
- ④ L1 → L2 → L3



- ① equation
- ② Test case, constraints
- ③ Dry run
- ④ test case → think
- ⑤ code
- ⑥ verify (C++ code)

[Basic - String] → [Recursion] → [ ]  
 ↑  
90% → (10%)  
 ↳ (L1 + L2)



0	1	4	2	8	2
4	3	6	5	0	4
1	2	4	1	4	6
2	0	7	3	2	2
3	1	5	9	2	4
2	7	0	5	1	1

	2	1	2	3
0	10	23	12	18
1	22	21	9	1
2	5	0	2	3
3	0	6	14	2

	0	1	2	3
0			28	15
1			19	1
2			5	3
3	29	17	2	

MCX 6/9/20

$N=4$   
 $n=4$   $d=2$   
 $g_C=1$   
 $g_T=9$

$$\begin{array}{l} 3 + 1 = 4 \\ 1 + 1 = 2 \end{array}$$

2	1	1
1	0	1
9	1	1

```

int64_t state; int getuint64(int i) {
    int64_t u;
    int n = 1;
    while (i > 0) {
        u = (u << 1) | (i > 0);
        i = i >> 1;
    }
    return u;
}

int64_t get_uint64(int i) {
    int64_t u;
    int n = 1;
    while (i > 0) {
        u = (u << 1) | (i > 0);
        i = i >> 1;
    }
    return u;
}

int main() {
    int i = 0;
    while (i < 1000000000) {
        int64_t u = get_uint64(i);
        if (u < 0) {
            continue;
        }
        int64_t u2 = u;
        for (int d = 1; d <= 10; d++) {
            int r = u % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u2 = u2 * 10 + r;
            }
            u2 = u2 * 10 + r;
        }
        int64_t u3 = u2;
        for (int d = 1; d <= 10; d++) {
            int r = u3 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u3 = u3 * 10 + r;
            }
            u3 = u3 * 10 + r;
        }
        int64_t u4 = u3;
        for (int d = 1; d <= 10; d++) {
            int r = u4 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u4 = u4 * 10 + r;
            }
            u4 = u4 * 10 + r;
        }
        int64_t u5 = u4;
        for (int d = 1; d <= 10; d++) {
            int r = u5 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u5 = u5 * 10 + r;
            }
            u5 = u5 * 10 + r;
        }
        int64_t u6 = u5;
        for (int d = 1; d <= 10; d++) {
            int r = u6 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u6 = u6 * 10 + r;
            }
            u6 = u6 * 10 + r;
        }
        int64_t u7 = u6;
        for (int d = 1; d <= 10; d++) {
            int r = u7 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u7 = u7 * 10 + r;
            }
            u7 = u7 * 10 + r;
        }
        int64_t u8 = u7;
        for (int d = 1; d <= 10; d++) {
            int r = u8 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u8 = u8 * 10 + r;
            }
            u8 = u8 * 10 + r;
        }
        int64_t u9 = u8;
        for (int d = 1; d <= 10; d++) {
            int r = u9 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u9 = u9 * 10 + r;
            }
            u9 = u9 * 10 + r;
        }
        int64_t u10 = u9;
        for (int d = 1; d <= 10; d++) {
            int r = u10 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u10 = u10 * 10 + r;
            }
            u10 = u10 * 10 + r;
        }
        int64_t u11 = u10;
        for (int d = 1; d <= 10; d++) {
            int r = u11 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u11 = u11 * 10 + r;
            }
            u11 = u11 * 10 + r;
        }
        int64_t u12 = u11;
        for (int d = 1; d <= 10; d++) {
            int r = u12 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u12 = u12 * 10 + r;
            }
            u12 = u12 * 10 + r;
        }
        int64_t u13 = u12;
        for (int d = 1; d <= 10; d++) {
            int r = u13 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u13 = u13 * 10 + r;
            }
            u13 = u13 * 10 + r;
        }
        int64_t u14 = u13;
        for (int d = 1; d <= 10; d++) {
            int r = u14 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u14 = u14 * 10 + r;
            }
            u14 = u14 * 10 + r;
        }
        int64_t u15 = u14;
        for (int d = 1; d <= 10; d++) {
            int r = u15 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u15 = u15 * 10 + r;
            }
            u15 = u15 * 10 + r;
        }
        int64_t u16 = u15;
        for (int d = 1; d <= 10; d++) {
            int r = u16 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u16 = u16 * 10 + r;
            }
            u16 = u16 * 10 + r;
        }
        int64_t u17 = u16;
        for (int d = 1; d <= 10; d++) {
            int r = u17 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u17 = u17 * 10 + r;
            }
            u17 = u17 * 10 + r;
        }
        int64_t u18 = u17;
        for (int d = 1; d <= 10; d++) {
            int r = u18 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u18 = u18 * 10 + r;
            }
            u18 = u18 * 10 + r;
        }
        int64_t u19 = u18;
        for (int d = 1; d <= 10; d++) {
            int r = u19 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u19 = u19 * 10 + r;
            }
            u19 = u19 * 10 + r;
        }
        int64_t u20 = u19;
        for (int d = 1; d <= 10; d++) {
            int r = u20 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u20 = u20 * 10 + r;
            }
            u20 = u20 * 10 + r;
        }
        int64_t u21 = u20;
        for (int d = 1; d <= 10; d++) {
            int r = u21 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u21 = u21 * 10 + r;
            }
            u21 = u21 * 10 + r;
        }
        int64_t u22 = u21;
        for (int d = 1; d <= 10; d++) {
            int r = u22 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u22 = u22 * 10 + r;
            }
            u22 = u22 * 10 + r;
        }
        int64_t u23 = u22;
        for (int d = 1; d <= 10; d++) {
            int r = u23 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u23 = u23 * 10 + r;
            }
            u23 = u23 * 10 + r;
        }
        int64_t u24 = u23;
        for (int d = 1; d <= 10; d++) {
            int r = u24 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u24 = u24 * 10 + r;
            }
            u24 = u24 * 10 + r;
        }
        int64_t u25 = u24;
        for (int d = 1; d <= 10; d++) {
            int r = u25 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u25 = u25 * 10 + r;
            }
            u25 = u25 * 10 + r;
        }
        int64_t u26 = u25;
        for (int d = 1; d <= 10; d++) {
            int r = u26 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u26 = u26 * 10 + r;
            }
            u26 = u26 * 10 + r;
        }
        int64_t u27 = u26;
        for (int d = 1; d <= 10; d++) {
            int r = u27 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u27 = u27 * 10 + r;
            }
            u27 = u27 * 10 + r;
        }
        int64_t u28 = u27;
        for (int d = 1; d <= 10; d++) {
            int r = u28 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u28 = u28 * 10 + r;
            }
            u28 = u28 * 10 + r;
        }
        int64_t u29 = u28;
        for (int d = 1; d <= 10; d++) {
            int r = u29 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u29 = u29 * 10 + r;
            }
            u29 = u29 * 10 + r;
        }
        int64_t u30 = u29;
        for (int d = 1; d <= 10; d++) {
            int r = u30 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u30 = u30 * 10 + r;
            }
            u30 = u30 * 10 + r;
        }
        int64_t u31 = u30;
        for (int d = 1; d <= 10; d++) {
            int r = u31 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u31 = u31 * 10 + r;
            }
            u31 = u31 * 10 + r;
        }
        int64_t u32 = u31;
        for (int d = 1; d <= 10; d++) {
            int r = u32 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u32 = u32 * 10 + r;
            }
            u32 = u32 * 10 + r;
        }
        int64_t u33 = u32;
        for (int d = 1; d <= 10; d++) {
            int r = u33 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u33 = u33 * 10 + r;
            }
            u33 = u33 * 10 + r;
        }
        int64_t u34 = u33;
        for (int d = 1; d <= 10; d++) {
            int r = u34 % d;
            if (r < 0) {
                continue;
            }
            if (r < 10) {
                u34 = u34 * 10 + r;
            }
            u34 = u34 * 10 + r;
        }
        int64_t u35 = u34;
        for (int d = 1; d <= 10; d++) {
            int r = u35 % d;

```