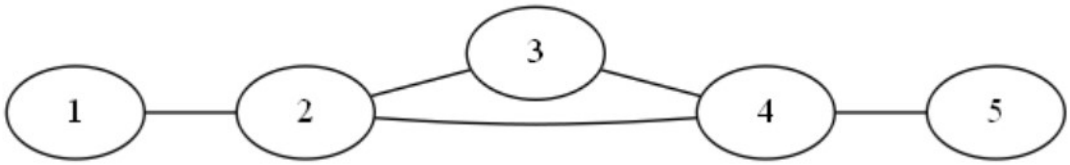
ABC 160D Line++

Given an undirected graph with n vertices and n edges of the following form: $n-1$ edges that connect vertices i and $i+1$, and an extra edge connecting vertices x and y ($x < y$). Find the number of pairs (a, b) such that $a < b$ and the shortest distance from a to b is k, for all $1 \leq k \leq n-1$. ($n \leq 2000$)

For example, if $n = 5, x = 2, y = 4$, the output should be $5\ 4\ 1\ 0$.



shortest distance from a to b is 1: $(1,2), (2,3), (2,4), (3,4), (4,5)$

shortest distance from a to b is 2: $(1,3), (1,4), (2,5), (3,5)$

shortest distance from a to b is 3: $(1,5)$

shortest distance from a to b is 4:

Try to think of a solution before reading on!

Since n is small, an $O(n^2)$ solution should pass. → We can just iterate for all pairs (a, b) and compute the shortest distance for each of them!

For each pair (a, b), you have 2 choices: use x, y edge, or don't use it.

If you don't use x, y edge, distance $= b - a$

if you use x, y edge, you need to go from a to x, then x to y, then y to b.

Therefore, distance $= |a - x| + 1 + |b - y|$

```
#include <cstdio>
#include <algorithm>
using namespace std;
int main(){
    int n, x, y;
    scanf("%d %d %d", &n, &x, &y);
    x--, y--;
    int c[n];
    fill(c, c + n, 0);
    for (int i = 0; i < n; i++){
        for (int j = i + 1; j < n; j++){
            c[min(j - i, abs(x - i) + 1 + abs(y - j))]++;
        }
    }
    for (int i = 1; i < n; i++) printf("%d\n", c[i]);
}
```