import java.util.Scanner;

public class HamsterColoring {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

long n = sc.nextLong();

long m = sc.nextLong();

int k = sc.nextInt();

long[] count = new long[k];

long totalDiagonals = n + m - 1;

for (int color = 0; color < k; color++) {

long firstD = color + 1;

if (firstD > totalDiagonals) continue;

long lastD = firstD + ((totalDiagonals - firstD) / k) \* k;

count[color] = sumDiagonalLengths(n, m, firstD, lastD, k);

}

for (long c : count) {

System.out.println(c);

}

}

private static long sumDiagonalLengths(long n, long m, long dStart, long dEnd, long step) {

long sum = 0;

for (long d = dStart; d <= dEnd; d += step) {

sum += diagonalLength(n, m, d);

}

return sum;

}

private static long diagonalLength(long n, long m, long d) {

if (d <= Math.min(n, m)) {

return d;

} else if (d > Math.max(n, m)) {

return n + m - d;

} else {

return Math.min(n, m);

}

}

}