Subset Sums

JRM

For many sets of consecutive integers from 1 through N (1 <= N <= 39), one can partition the set into two sets whose sums are identical.

For example, if N=3, one can partition the set {1, 2, 3} in one way so that the sums of both subsets are identical:

- {3} and {1,2}

This counts as a single partitioning (i.e., reversing the order counts as the same partitioning and thus does not increase the count of partitions).

If N=7, there are four ways to partition the set {1, 2, 3, ... 7} so that each partition has the same sum:

- {1,6,7} and {2,3,4,5}
- {2,5,7} and {1,3,4,6}
- {3,4,7} and {1,2,5,6}
- {1,2,4,7} and {3,5,6}

Given N, your program should print the number of ways a set containing the integers from 1 through N can be partitioned into two sets whose sums are identical. Print 0 if there are no such ways.

Your program must calculate the answer, not look it up from a table.

PROGRAM NAME: subset

INPUT FORMAT

The input file contains a single line with a single integer representing N, as above.

SAMPLE INPUT (file subset.in)

7

OUTPUT FORMAT

The output file contains a single line with a single integer that tells how many same-sum partitions can be made from the set {1, 2, ..., N}. The output file should contain 0 if there are no ways to make a same-sum partition.

SAMPLE OUTPUT (file subset.out)

4