

Problem A: Adaptive

KKU has introduced new traffic lights featuring an adaptive control system, designed to optimize traffic management for increased safety and efficiency.



Unfortunately, after several months of observation, the system's “**adaptive**” capabilities have not met expectations. Your task is to write a program to help improve traffic management.

The map of KKU is represented as a $R \times C$ grid. Each cell within this grid corresponds to a traffic light, and each traffic light has its own waiting time. We would like to set the waiting time for each traffic light to satisfy the following conditions:

- The waiting time of each traffic light must be a positive integer.
- The shortest total time required to travel from the top-left cell to the bottom-right cell must be exactly T_A
- The shortest total time required to travel from the top-right cell to the bottom-left cell must be exactly T_B

However, it may not always be possible to satisfy all these conditions. Your program must determine for which scenario it is possible to do so.

Input:

The first line of input is an integer T , representing the number of test cases. Each test case consists of a single line containing four numbers: R , C , T_A and T_B .

Output:

For each test case, print “Adaptive” if it is possible to set up the traffic system to satisfy those conditions; otherwise, “No”.

Sample Input	Sample Output
3 2 2 120 100 3 3 10 11 3 4 2 1	Adaptive Adaptive No

Explanation:

- Test case #1: it is possible to design the traffic system as follows

40	40
20	60

There are two paths from the top-left cell to the bottom-right cell:
(40 → 40 → 60 = 140) and (40 → 20 → 60 = **120**)

There are two paths from the top-right cell to the bottom-left cell:
(40 → 40 → 20 = **100**) and (40 → 60 → 20 = 120)

- Test case #2: it is possible to design the traffic system as follows

1	3	2
12	1	2
5	1	4

The shortest path from the the top-left cell to the bottom-right cell:
(1 → 3 → 1 → 1 → 4 = **10**)

The shortest path from the top-right cell to the bottom-left cell:
(2 → 2 → 1 → 1 → 5 = **11**)

Constraints:

- $1 \leq T \leq 50$
- $1 \leq R, C \leq 100, 1 \leq T_A, T_B \leq 1,000,000$