I would reject the provided test cases for the following reasons:

- The last test case adds 1 to the -MAX_INT which would actually give an answer and not underflow exception.
- Since the last test case is invalid, there is no test case for underflow.
- There are a few test cases that can be added to the test suite to make it comprehensive.

Addition to the given test cases (omitting the last one from the original)

Unit Name	Adder	
Summary (e.g. Interface spec)	Takes two integers as input and returns their sum (as an integer)	
Input Type(s)	Input Value(s)	Expected Result
int, int	MIN_INT, -1	Underflow Exception
int, int	-1, 1	0
int, int	0, 0	0
int, int	-1, 0	-1
int, int	0, 1	1
int, int	1, 0	1

A few test cases added makes sure that there is an Underflow, the order of the arguments shouldn’t affect the result, and same valued and opposite numbers results to zero.