Mohit Sharma (2019CS10372) Vivek Choudhary (2019CS10413)

Approach:

1) Input length of string & string itself.
2) After this traversed the string character by character. If the character is not '+', '-', '*' or a digit, then error is thrown.
3) An array is made for the stack operations.
   (i) If a digit is encountered while traversing the string, then it is pushed in the array and the counter to array is incremented.
   (ii) When either of '*', '-', '+' is encountered, two characters from end of array are taken and counter is decreased. If two characters are not present than error is thrown otherwise the corresponding operation is done in correct order. Overflow is detected and error is thrown is such case, otherwise the result is pushed in stack and counter to array is incremented.

   The counter to array points at a location where a new element is to be inserted.
4) At the end, if one element is present in array than it is printed as result otherwise error is thrown.

Vivek Choudhary, Mohit Sharma

Test cases

① n = 1, exp = - => error ✓

② n = 3, exp = 12-* => -1 ✓

③ n = 5, exp = 123+* => 5 ✓

④ n = 101, exp = 1111...11 (51) +++++...+ (50) => 51 ✓

⑤ n = 11, exp = 123456+*+*+ => 95 ✓

⑥ n = 11, exp = 12345789++ => error ✓

⑦ n = 7, exp = ++*+**+ => error ✓

⑧ n = 1, exp = 1 => 1 ✓

⑨ n = 21, exp = 9999999999 (11 times) ********** (9 times) + => calculation overflow ✓

⑩ n = 5, exp = +---- => invalid expression ✓

⑪ n = 7, exp = 779++92 => invalid expression ✓