**1.6 Algorithm:**

Step 1: Initiating a variable x with value 0.1.

Step 2: Initiating another variable prev\_ans with value 0.

Step 3: A variable ans is defined to store the answer in each step starting from 0.

Step 4: A variable prev\_ans is defined to store the answer in previous step starting from 0.

Step 4: Creating a For loop for calculating in each expansion term of the Taylor series of ln(1+x) format.

Step 4.1: For loop is initiated with value i. For loop is calculated until number of iterations becomes 100.

Step 4.2: prev\_ans is assigned the value of ans.

Step 4.3: i is checked for even or odd and accordingly the value is subtracted/added for each term in the Taylor series respectively.

Step 4.4: Another if condition is given to check the difference between previous value and present value. If it is same up to 5 decimal places i.e. difference less than 10-5, the for loop is broken and the step comes outside of the loop.

Step 4.5: If 4.4 is not satisfied i is updated with i = i+1, return to step 4.

Step 5: Outputting the value of ln (1.1) i.e., ln (1+0.1) with tolerance 5 decimal place and maximum iteration 100 whichever is achieved earlier.

**Discussion:**

We know that the expansion of ln(1+x) is given by Taylor’s series which come out to be .

We were required to find out ln (1+0.1) up to a precision of 5 decimal places. We have ensured this by keeping a condition of checking the value of increment/decrement of each term of Taylor series. If the value of the increment/decrement is less that 10-5  then we have terminated the loop. Hence achieving the target of 5 decimal places. We have also introduced a maximum iteration of 100 of the loops. In case the desired loop is not achieved so this condition will ensure the loop does not enter a state of infinite iterations and the value is printed accordingly.