|  |  |
| --- | --- |
| Ex.No:2 | Finding Exponentiation (Power of an Number) |
|  |

***Aim:***

To develop a python program to finding exponentiation (power of a number).

***Algorithm:***

***Main Algorithm***

|  |  |
| --- | --- |
| **Step 1:** | Start Process |
| **Step 2:** | Get a number from user to find exponentiation value |
| **Step 3:** | Assign result as zero and i as 0 |
| **Step 4:** | Call Power Finder with number and i value and store the return value into power\_value |
| **Step 5:** | Call Fact Finder with i value and store the return value in fact\_value |
| **Step 6:** | Compute power\_value divided by fact\_value and store in current\_term |
| **Step 7:** | Accumulate current\_term with result |
| **Step 8:** | Increment i |
| **Step 9:** | If i is less than 10 then goto Step 4 |
| **Step 10:** | Else goto Step 11 |
| **Step 11:** | Print result |
| **Step 12:** | Stop Process |

***Power Finder Algorithm***

|  |  |
| --- | --- |
| **Step 1:** | Start Process |
| **Step 2:** | Receive number and power |
| **Step 3:** | Assign result as 1 and i as 0 |
| **Step 4:** | Compute multiplication with result and number |
| **Step 5:** | Accumulate the value with result |
| **Step 6:** | If i is less than power then goto Step 4 |
| **Step 7** | Else return result |
| **Step 8:** | Stop Process |

***Factorial Finder Algorithm***

|  |  |
| --- | --- |
| **Step 1:** | Start Process |
| **Step 2:** | Receive number |
| **Step 3:** | If number is equals to 1 then return 1 |
| **Step 4:** | Else compute number \* Call the Factorial Finder with number – 1 and return the value |
| **Step 6:** | Stop Process |

***Flow Chart:***

***Pseudo Code:***

START

READ number

SET guess = 1

WHILE TRUE

COMPUTE fx=guess \* guess – number

COMPUTE dx=2 \* guess

COMPUTE actual = guess – (fx/dx)

IF actual is equal to guess THEN

PRINT “Square root value of given number ” number “ is ” actual

ELSE

ASSIGN guess as actual

END IF

END WHILE

STOP

***Program:***

*newtonmethod.py*

print("Enter the number to find the root");

number=int(input())

guess = 1

while True:

fx = guess\*guess - number

dx = 2\*guess

actual = guess - ((fx)/dx)

actual = round(actual,4)

if guess == actual:

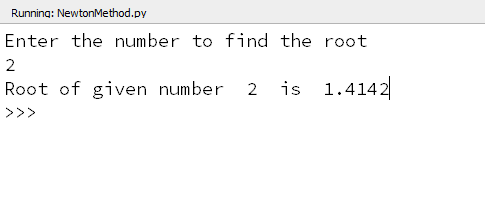
print("Root of given number ",number," is ",actual)

break

else:

guess = actual

***Output***

**

***Result:***

Thus the program to find the square root of the given number using Newton’s iterative method was developed and tested successfully.