Long long int max one can only sore up to 10 ^18 number not more than that.

Sol:

// C++ program to compute factorial of big numbers

#include<iostream>

using namespace std;

// Maximum number of digits in output

#define MAX 500

int multiply(int x, int res[], int res\_size);

// This function finds factorial of large numbers

// and prints them

void factorial(int n)

{

int res[MAX];

// Initialize result

res[0] = 1;

int res\_size = 1;

// Apply simple factorial formula n! = 1 \* 2 \* 3 \* 4...\*n

for (int x=2; x<=n; x++)

res\_size = multiply(x, res, res\_size);

cout << "Factorial of given number is \n";

for (int i=res\_size-1; i>=0; i--)

cout << res[i];

}

// This function multiplies x with the number

// represented by res[].

// res\_size is size of res[] or number of digits in the

// number represented by res[]. This function uses simple

// school mathematics for multiplication.

// This function may value of res\_size and returns the

// new value of res\_size

int multiply(int x, int res[], int res\_size)

{

int carry = 0; // Initialize carry

// One by one multiply n with individual digits of res[]

for (int i=0; i<res\_size; i++)

{

int prod = res[i] \* x + carry;

// Store last digit of 'prod' in res[]

res[i] = prod % 10;

// Put rest in carry

carry = prod/10;

}

// Put carry in res and increase result size

while (carry)

{

res[res\_size] = carry%10;

carry = carry/10;

res\_size++;

}

return res\_size;

}

// Driver program

int main()

{

factorial(100);

return 0;

}