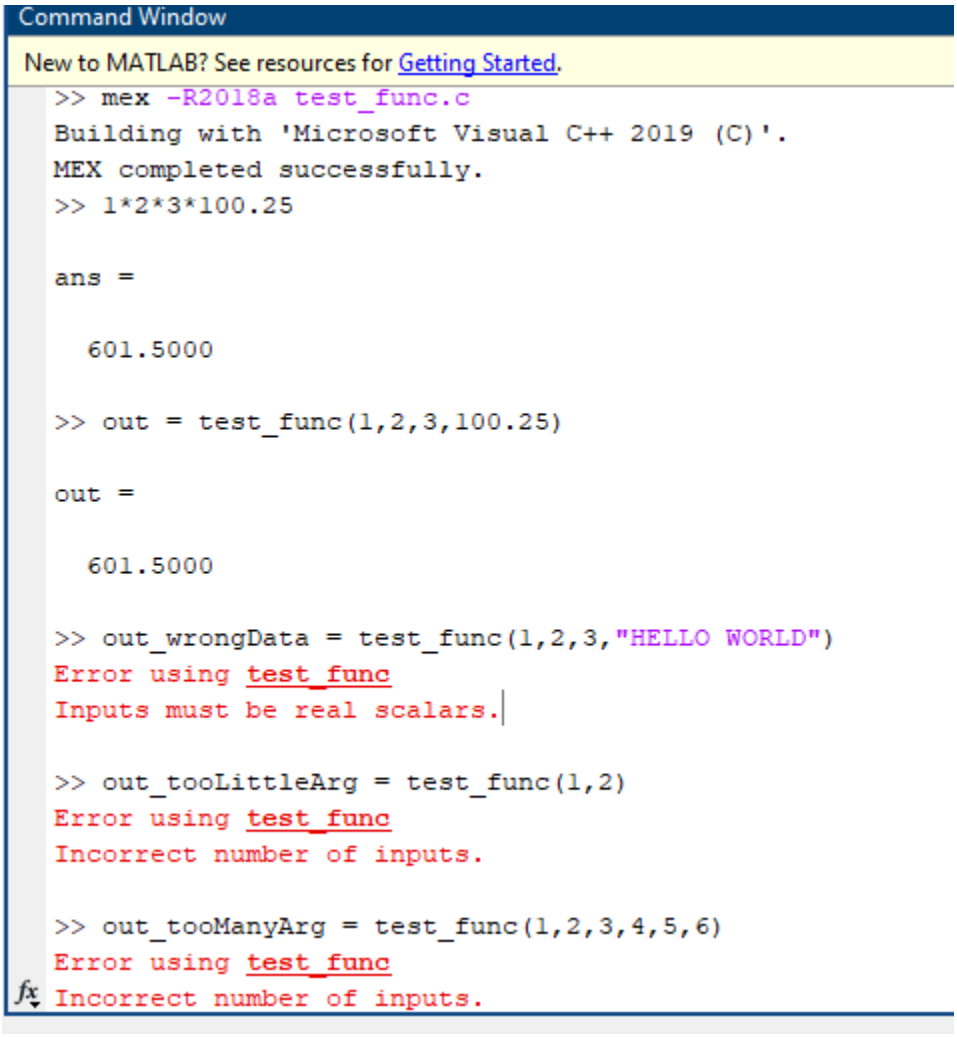
#1: Make MATLAB C CODE (MEX) function that takes 4 input arguments and calculates their product. Use filename test_func.c. Verify in MATLAB command window that its working (write out = test_func(1,2,3,100.25) and verify that you get the correct output). Include the output in your report. Try also to feed the function wrong data types such as out = test_func(1,2,3,"HELLO WORLD") // what happens now? Try feeding too little or too many input arguments, what happens now?

Answer:

I have added here Matlab command lines and also the results. Besides, I have added the screenshots of the command line's answer. I have tried to get the result according to the instructions for the first task and I got it. All the answers and commands are given below. Firstly, I completed the MEX building successfully after making the test_func.c for multiplying the 4 inputs. Besides, I put the header file in the same directory. Secondly, I put the wrong input “Hello word” but it showed the error because we declared the double variable. Finally, I have tried increasing and decreasing the number of input values but got the error because we fixed the 4 inputs in our test_func .c file.

```
>> mex -R2018a test_func.c
Building with 'Microsoft Visual C++ 2019 (C)'.
MEX completed successfully.
>> 1*2*3*100.25

ans =

  601.5000

>> out = test_func(1,2,3,100.25)

out =

  601.5000

>> out_wrongData = test_func(1,2,3,"HELLO WORLD")
Error using test_func
Inputs must be real scalars.

>> out_tooLittleArg = test_func(1,2)
Error using test_func
Incorrect number of inputs.

>> out_tooManyArg = test_func(1,2,3,4,5,6)
Error using test_func
```

Incorrect number of inputs.



#2: Make an integer (int) array of 7 elements [1,2,3,4,100, 256, 300] and calculate the sum of even (divisible by 2) values in the array by while loop. Print the sum using printf and verify that you get the correct output. Hint: Use the modulo/remainder operator [“%”]. Include the output in your report. [Use Standard C, run for example using the online C compiler at https://www.onlinegdb.com/online_c_compiler]

Answer:

For task 2, I have made the 7 elements array and counted the only even number using the remainder operator. After that, I made the summation of all even numbers using a while loop. And I have verified the answer many times by changing the array’s value. I have added here the code and also the results. Besides, I have added the .c file in the Zip folder.

```c
#include <stdio.h>
#include <stdlib.h>
int main()
{
   int arr[] = {1,2,3,4,100, 256, 300};
   int size_arr = sizeof(arr)/sizeof(arr[0]); // find out the size of array
   // initilization
   int i = 0;
   int arr_sumEven = 0;

   while(i < size_arr) // condition check
   {
      if (arr[i]%2 == 0) // even number calculation
      {
         arr_sumEven = arr_sumEven + arr[i];

      }

      i = i + 1;

   }

   printf("%d", arr_sumEven);  // printing the output

   return 0;
}
```

#3: Repeat task #2 but fill the integer array of 7 elements by random numbers between 0 and 100 (generated with help of rand() function from). Hint: https://www.tutorialspoint.com/c_standard_library/c_function_rand.htm You do not need to use srand to seed the pseudo-random generator (you can if you want), so pseudo-random sequence will be the same every time you run the program.

Answer:

In task three, Basically I have done the same procedure as in task 2 but I have just generate 7 random values for making the array using the rand() function. And I have tested with and observed the correct result. In the zip folder, I have added the .c file. In addition, I have attached here the well documented code and the result also.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int array_size_random = 7;              // number of array size
    int arr[array_size_random],k;           // initilization
    int size = sizeof(arr)/sizeof(arr[0]);  // calculate the size of array
   for(k=0;k<size;k++)
   {
      arr[k]=rand()%100;                  //Generate number between 0 to 99
   }

   printf("\nElements of the array::");
   for(int m=0;m<size;m++)                  // print the array nElements
   {
      printf("\nElement number %d::%d", m, arr[m]);
   }


    //int arr[] = {1,2,3,4,100, 256, 300};
    int size_arr = sizeof(arr)/sizeof(arr[0]); // find out the size of array
    // initilization
    int i = 0;
    int arr_sumEven = 0;

    while(i < size_arr) // condition check
    {
        if (arr[i]%2 == 0) // even number calculation
        {
            arr_sumEven = arr_sumEven + arr[i];

        }

        i = i + 1;

    }
```

```
    printf("\nsummation of even number: %d", arr_sumEven);  // printing the output

    return 0;
}
```