**Practice on 1D Array & Function**

1. Write a C program to calculate sum of factorial of odd numbers upto *n* (take *n* as input, it will be a positive integer). Write a function for factorial calculation, which will receive an integer and return its factorial.

|  |  |
| --- | --- |
| Sample Input | Sample Output |
| 5 | 127 |
| 4 | 7 |

1. Given three non negative integers, *n*, *m*, *p* calculate *nm*% *p* using a C function.

|  |  |
| --- | --- |
| Sample Input | Sample Output |
| 5 2 100 | 25 |
| 4 3 11 | 9 |

1. Given some numbers find their median. Take a non negative integer, *n,* followed by n real numbers as input. Find their median (the middle value of the sorted elements if there are odd number of elements and the average of two middle values in case of even number of elements). Do not use any separate function to find median.

|  |  |
| --- | --- |
| Sample Input | Sample Output |
| 5  4 3 2 5 4 | 4 |
| 4  5 8 2 4 | 4.5 |

1. Solve problem 3 again, but this time use a function to calculate median. The function should receive an integer array and its length as parameter and return the median. The array passed to the function should remain unchanged after the function is called.
2. Given an integer *n*, print it as product of its prime factors with appropriate powers. Then you have to integer numbers as input continuously until you get -1 and if the integeris a prime factor of *n* print its power in the prime factorization of *n*. All the inputs will be in [2, 10000] range.

|  |  |
| --- | --- |
| Sample Input | Sample Output |
| 36  2  5  3  -1 | 2^2 X 3^2  2  0  2 |
| 1176  7  2  -1 | 2^3 X 3^1 X 7^2  2  3 |