Operating Systems

Lab assignment 1

26 September 2018

Header files for you

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <string.h>
```

Q1) Write a C program that performs the following task (6 marks) :

1. Take n from the user and declare a 1D array with size n and fill it with user inputs.

2. Create 2 threads (t_even and t_odd) that work **IN PARALLEL** such that
   -> t_even calculates sum of elements present at even indices
   -> t_odd calculates sum of elements present at odd indices

   * **Both threads should call a common function**

3. Print the outputs calculated by t_even and t_odd threads as your answer.

Input format :
1 st line contains 1 integer n (1<= n <= 10^5) which represents size of array
2nd line contains 'n' space separated integers representing elements of the array

Output format :
output sum of threads t_even and t_odd

Sample input:

```
7
2 0 1 6 7 8 6
```

Sample output:

```
t_even got sum = 16
t_odd got sum = 14
```

Q2) Write a C program that performs the following task (18 marks) :

1. Take 'n' and 'm' as input from user where n and m are dimensions of a 2 D matrix.

2. Create n threads such that the i th thread sorts the i th row of the matrix. (**Threads should be compulsorily executed in parallel**)

3. Create a median array which stores median of the i th row in its i th element

4. Print the final sorted matrix and the median array as your final answer.

NOTE :

- You can use any sorting algorithm (O (n^2) also accepted)
- Your code shall be evaluated by test case files uploaded by us towards the end.

Input format :

1st line contains a single integer n representing number of rows
2nd line contains a single integer m representing number of columns
Following n lines contains m space seperated integers (each line representing elements of a row)

Output format :

Print Row sorted matrix (size n*m)
Print Median array (size n)

Sample testcases

Sample input 1

```
3
4
3 6 7 5
3 5 6 2
9 1 2 7
```

Sample output 1

```
Row sorted matrix
3 5 6 7
2 3 5 6
1 2 7 9

Median array
5.5 4.0 4.5
```

Sample input 2

```
4
3
3 6 7
5 3 5
6 2 9
1 2 7
```

Sample output 2

Row sorted matrix
3 6 7
3 5 5
2 6 9
1 2 7

Median array
6.0 5.0 6.0 2.0