1. Given an array of 1 million integers, use C and pthread to write a parallel program to find out the sum of the array and the second maximum. Assume the entire array is stored initially in one location and is distributed to the different threads for parallel processing.
You can initialize the array using random values, using rand()

2. Write a program using C and pthreads to perform a parallel matrix multiplication routine. The goal is to multiply an MxN matrix called A by an NxP matrix called B and then store the result into the MxP matrix called C. You should design your program in such a manner that each thread will have the threads run in a loop that computes single rows *(not for each element)* of the result, C.

3. Estimate pi using the Maclaurin series for arctan(x):

4.4 The Maclaurin series expansion for the arctangent of x is de-fined for $|x| \le 1$ as

$$\arctan x = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} x^{2n+1}$$

Since arctan(1) = pi/4, we can compute
pi = 4*[1 - 1/3 + 1/5 - 1/7 + 1/9 - . . .]
Run your program as pi_mutex <number of threads> <n>

- pi_mutex is the executable of your code
- n is the number of terms of the Maclaurin series to use
- n should be evenly divisible by the number of threads and be greater than 100,000

Output: The estimate of pi using multiple threads.