Note: Create a subdirectory in your home directory and call it `<your_last_name>.3`, where `<your_last_name>` is your real last name (for example, bhatia for me). Do all the programs for this assignment in that directory. After you are done, submit the code by typing the following command:

```
~sanjiv/bin/handin <your_last_name>.3 cs278 3
```

Again, do not forget to substitute for `<your_last_name>`. The command should be executed from your home directory. Do the assignment on hoare.

1. Empirically determine the effect of removing the test `if a[i]` from Program 3.5 (in your textbook), for \( N = 10^3, 10^4, 10^5 \), and \( 10^6 \).

2. Write a program that checks whether a given string of characters is a palindrome.

3. Write a recursive program to compute \( \log(N!) \). Modify it to compute \( N! \mod M \), such that overflow is no longer an issue. Try running your program for \( M = 997 \) and \( N = 10^3, 10^4, 10^5 \), and \( 10^6 \), to get an indication of how your programming system handles deeply nested recursive calls.

4. Write two functions to compute factorial, one iterative and the other recursive. Call them from a program and give the time in a tabular form on how long it takes to compute the factorial by two methods for different numbers (use a for loop to try different values of number whose factorial is to be computed). Use the function `gethrtime(3)` on Unix to get time in microseconds for comparison. Call this function just before and just after the call to your factorial functions.