Due Date: 30.10.2016, 23:55

CENG 313 – Operating Systems

Homework #1

Implement a C program that is explained below using UNIX operating system.

In this homework we want you to implement a multi-threaded program that creates a file and performs a word (sub-string) search and count program. First your program should create a process and your process should create multiple threads. The tasks of the threads can be separated into 3 main tasks.

1. Create a file
2. Write the file (fill in the file with random letters without blanks)
3. Search and count the given **sub-strings**.

Detailed Explanation of Tasks

1. Create a file

Your program should create an empty file by a thread. Name the file as "**big.txt**".

2. Write the file

Your program should fill in the file by a thread with random letters. The letters can be lower or upper case ([a-zA-Z]). Your file must be filled with 250.000.000 letters that are concatenated to each other. The file should not contain Turkish characters/letters, only English characters are allowed.

Example of file content

".....

lfnqopSrehglaKQIAksdlfkahQWMNsldfkqMerhPQIBMAqnlgkaIASEGKCXJavjqovAJGi

KnvhajwIGJahgIqAVHBqhvahsJqbJZubENguaOSasdiQNvuasJhasqHMNKzorBJAaarEbh

....."

3. Search and count the given sub-strings

Your thread/s should search for 5 sub-strings with different sizes. The sub-strings you are responsible to find are;

- x
- os
- cpu
- disk
- cache

However, your search shouldn't be case sensitive. If a sub-string "os/OS/Os/oS" is found, you should increment the sub-string count for "os".

Run your program with different **numbers of threads** and **how many of the threads are responsible of a specific task**. Running your program with only "n" number of threads will not gain you points.

You should submit your homework as a zip file named "name_surname_studentID.zip" with a ".c" file and a document that includes;

- A table that shows the execution time of each configuration (how many threads are used for tasks). And your comments on the results.

And answers to these questions;

- What is the optimal number of threads for your computer? Why do you think it is the optimal number?
- When the use of threads does become necessary or unnecessary? Why? Explain your answer.

HINT

- Before you creating your "**big.txt**", try working with a smaller file. Therefore, you can ensure that your search and counting algorithms are working properly.

ASSIGNMENT RULES!

- Cheating will **NOT** be tolerated!
- For any detected cheating will be **graded as 0.**
- Late Submissions will not be allowed.

GRADE REDUCTIONS

Since you are Junior students you are expected that you are aware of; error handling, controls, software design etc. Please code your programs wisely. Possible grade reductions,

- **Lack of comment usage!**
- **Missing controls!**
- **No error handling!**
- **Unused/dead codes!**
- **Naming conventions!**

Please do not discuss with us why your grades decreased just because you have done the programming sins listed above!

NOTE: Do not ask from us about the possible errors that could occur. From this lecture and labs, you are expected to be aware of the possible errors.