Due Date: 29.12.2017

CENG 313 – Operating Systems
Homework #3

In this homework you are expected to cover two concepts of OS given below;

- Processes
- IPC (Pipes)
- Scheduling

Implement a C program that is explained below using WinAPI. You should implement a multi-process program that schedules processes by Shortest Job First (SJF). There should be 5 processes in your program. Each process has their own CPU burst time. The initial estimated value of CPU burst (τ_0) are given below:

Process #1: 300 ms
Process #2: 220 ms
Process #3: 180 ms
Process #4: 45 ms
Process #5: 255 ms

To calculate the predicated value for next CPU burst you should use the formula given below. The α value is set to 0.5. To calculate each process's actual length of nth CPU burst, you should assign a random value for each execution time. The random value should vary between 50 ms ≤ t_n≤ 300 ms.

1. $t_n =$ actual lenght of n^{th}CPU burst
2. $\tau_{n+1} =$ predicted value for the next CPU burst
3. $\alpha, 0 \leq \alpha \leq 1$
4. Define :

$$\tau_{n+1} = \alpha\, t_n + (1-\alpha)\tau_n .$$

Program should run the program 5 times. At each run you should calculate the new predicted value for the next CPU burst (t_{n+1}) and you should schedule your processes according to these burst times. Therefore, you should schedule your processes 5 times consecutively.

After each scheduling print the execution order with their t_n, τ_{n+1} and τ_n. Consider the 5 processes as child process, when the child process starts and ends it should notify the parent process and with a pipe.

EXAMPLE OUTPUT:

First Execution Order <*P4, P3, P2, P5, P1*>

(On parent process screen)

P4 started.
P4 ended.
P3 started.
P3 ended.
P2 started.
P2 ended.
P5 started.
P5 ended.
P1 started.
P1 ended.

Process	τ_n	t_n(Actual Length)	τ_{n+1}
P4	**45**	**-**	**-**
P3	**180**	**-**	**-**
P2	**220**	**-**	**-**
P5	**255**	**-**	**-**
P1	**300**	**-**	**-**

(NOTE: At this point, Actual lengths are calculated randomly)

Process	τ_n	t_n(Actual Length)	τ_{n+1}
P4	**45**	**70**	**-**
P3	**180**	**180**	**-**
P2	**220**	**150**	**-**
P5	**255**	**250**	**-**
P1	**300**	**100**	**-**

(NOTE: At this point, for each process τ_{n+1} values are calculated)

Process	τ_n	t_n(Actual Length)	τ_{n+1}
P4	**45**	**70**	**57.5**
P3	**180**	**180**	**180**
P2	**220**	**150**	**185**
P5	**255**	**250**	**257.5**
P1	**300**	**100**	**200**

Second Execution Order *<P4, P3, P2, P1, P5>*

(On parent process screen)

```
P4 started.
P4 ended.
P3 started.
P3 ended.
P2 started.
P2 ended.
P1 started.
P1 ended.
P5 started.
P5 ended.
```

Process	τ_n	t_n(Actual Length)	τ_{n+1}
P4	**57.5**	**-**	**-**
P3	**180**	**-**	**-**
P2	**185**	**-**	**-**
P1	**200**	**-**	**-**
P5	**257.5**	**-**	**-**

continues…

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 handlings, controls, software design etc. This lecture should be taken seriously and will take a crucial part in your work lives. 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.