REAL TIME EMBEDDED SYSTEMS

First Assignment: Implementation of Producers – Consumers Problem

Author: Portokalidis Stavros, *A.E.M.*: 9334, *email*: stavport@ece.auth.gr
Code link: https://github.com/sportokalidis/producers-consumers

Problem Description

The "Producers - Consumers" is a well-known programming problem. On the one hand, we have the producers, who add objects in a queue, and on the other hand there are consumers, who delete the objects from this queue. In this assignment, we are called to implement a parallel version of this problem. We have "p" producer threads and "q" consumer threads, who add and remove workFunction objects from queue buffer, respectively. WorkFunction objects include a function pointer and the arguments of this function. Consumers, when, remove an object from the queue buffer, must run this function. The target of this assignment is to take some statistics from remaining time of objects in the queue and find the suitable number of consumers that minimize the average remaining time.

Project and Code Explanation

https://github.com/sportokalidis/producers-consumers/blob/master/README.md

Hardware

4 x Intel Core i7-7500 CPU 2,70 GHz

Results

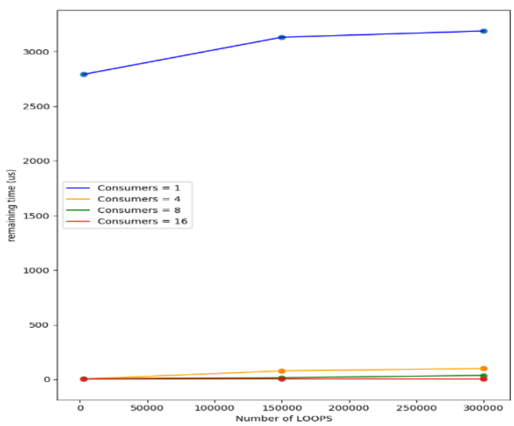
1st group of statistics: In this part, Queue size and number of producers are constant.
I) Queue_size = 1,000 and P = 4, **II)** Queue_size = 5,000 and P = 4, **III)** Queue_size = 20,000 and P = 4

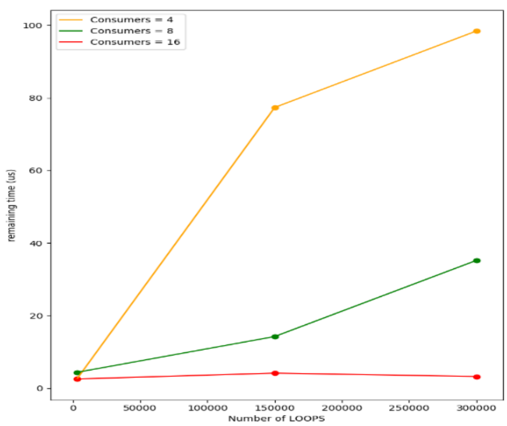
I)		Number of Consumers			
		1	4	8	16
Num of Loops	3,000	2793	2.52	4.36	2.52
	150,000	3132	77.36	14.22	4.14
	300,000	3189	98.43	35.21	3.22

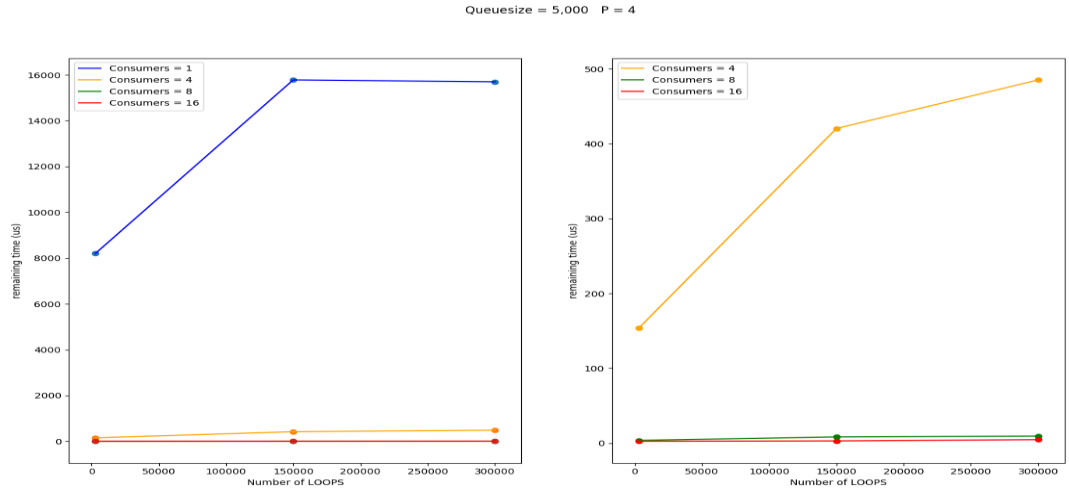
II)		Number of Consumers			
		1	4	8	16
Num of Loops	1,000	8205	15.35	3.68	2.49
	150,000	15780	42.04	83.3	2.94
	300,000	15699	48.53	9.4	4.629

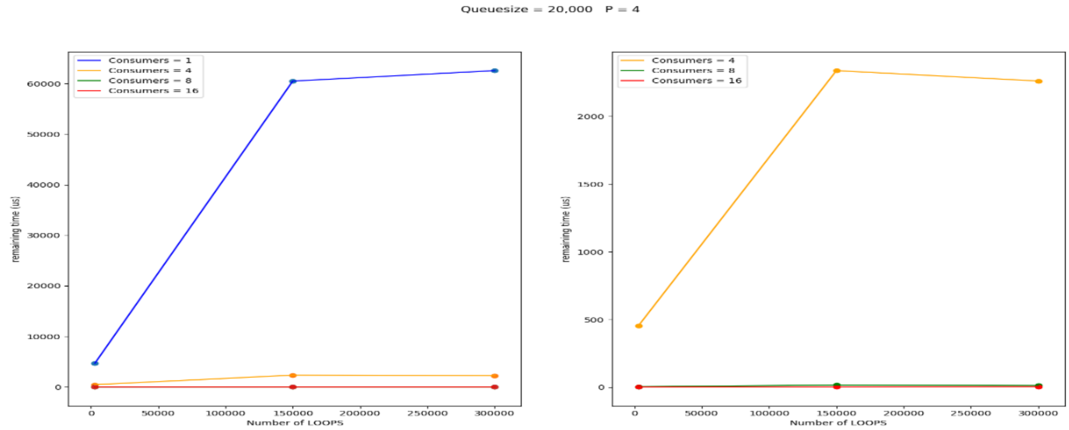
III)		Number of Consumers			
		1	4	8	16
Num of Loops	3,000	4675	454	4,7	2.5
	150,000	60494	2336	16.9	3.1
	300,000	62566	2260	13.05	4.5

Queuesize = 1,000 P = 4





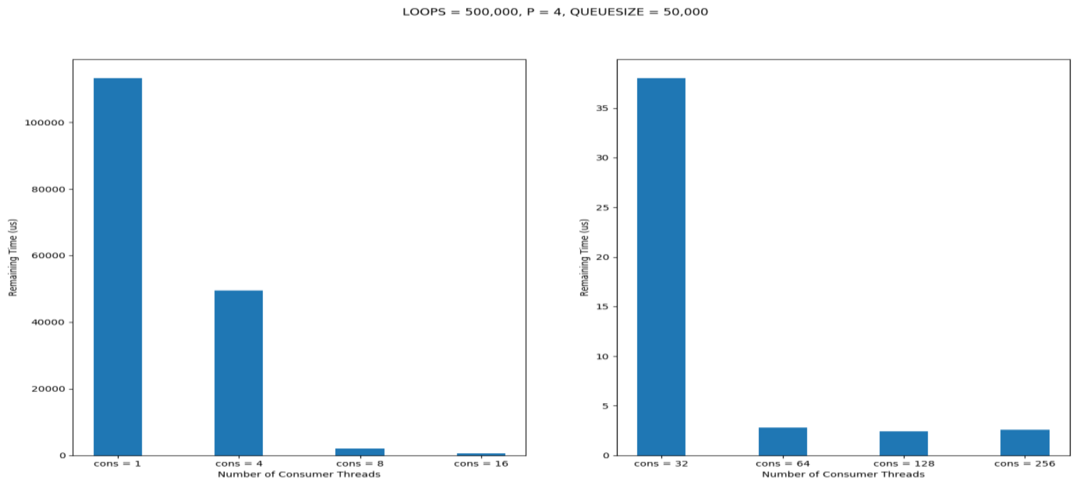




In conclusion, we understand that, for small values of Queue size and incoming packages, the difference in average remaining time is small for all consumers' numbers. However, when the number of Queue size and incoming packages increases, the difference in remaining time increases too.

<u>2nd part of statistics:</u> In this part, Queue size, number of producers and loops are constant. Now, we change the number of consumers in order to find the suitable number of treads which minimize the remaining time.

	Number of producers							
	Q = 1	Q = 4	Q = 8	Q = 16	Q = 32	Q = 64	Q = 128	Q = 256
Remaining time	113,263	49,539	2129	625	38	2.81	2.4	2.5



In conclusion, we understand that for big values of Loops and Queue size, the remaining time of objects in queue buffer decreases as much as the number of consumers' thread increases. From 2nd part of statistics and the graph, we notice that the number of threads that we will need to optimize the system is 64.