Q1. Write a program in C to simulate the behaviour of a priority-based scheduling algorithm. You can consider atmost 8 processes in the queue. You should try using structures to solve this question. Assume that all processes arrived at time zero. You can take the burst time for processes as random.

Q2. Repeat the previous question considering random arrival times of the processes. Further, you have to consider a pre-emptive algorithm.

Q3. Solve the third question of Tutorial 4 in C. This time, however, a user can enter the task number (T1, T2 or T3) and the instance number. In Tut 4, recall you were supposed to compute the completion time of the first instance of T3. Here, I can ask you to compute the completion time of x^{th} instance of T_n

For your convenience, the question is as follows:

Consider a uniprocessor system executing three tasks T1, T2 and T3, each of which is composed of an infinite sequence of jobs (or instances) which arrive periodically at intervals of 3, 7 and 20 milliseconds, respectively. The priority of each task is the inverse of its period and the available tasks are scheduled in order of priority, with the highest priority task scheduled first. Each instance of T1, T2 and T3 requires an execution time of 1, 2 and 4 milliseconds, respectively. Given that all tasks initially arrive at the beginning of the 1st milliseconds and task pre-emptions are allowed, the x^{th} instance of tasks T_n completes its execution at the end of _______________ milliseconds. x and n will be entered by the user.