National Institute of Technology Calicut
Department of Computer Science and Engineering
Third Semester B. Tech.(CSE)
CS2092D Programming Laboratory
Assignment #6 Modification (13-10-2023)

Naming Conventions for Submission

- The source codes must be named as

$$\texttt{ASSG<NUMBER>\_<ROLLNO>\_<FIRST-NAME>\_MOD.c}$$

(For example: $ASSG6\_BxxyyyyCS\_LAXMAN\_MOD.c$).

Standard of Conduct

- Violation of academic integrity will be severely penalized. Each student is expected to adhere to high standards of ethical conduct, especially those related to cheating and plagiarism. Any submitted work MUST BE an individual effort. Any academic dishonesty will result in zero marks in the corresponding exam or evaluation and will be reported to the department council for record keeping and for permission to assign F grade in the course. The department policy on academic integrity can be found at: http://cse.nitc.ac.in/sites/default/files/Academic-Integrity_new.pdf.

General Instructions

- Programs should be written in C language.

- Check your programs with sufficiently large values of inputs with in the range as specified in the question.

- Global and/or static variables should not be used in your program.

QUESTION

1. Write a menu driven program to design a job dispatcher. The job dispatcher selects a job to be executed from a pool of jobs based on its priority. Each job has a unique job-id JID ($1 \le JID \le 100$) and a priority p ($1 \le p \le 10$). Only one job can be executed at a time and the pool of jobs can get updated with more jobs. The new jobs, on its arrival, are added to the pool of jobs and are removed from the pool when they are scheduled for execution. From the pool, the job dispatcher selects the job with the highest priority for execution. The executed job is removed from the pool. Use priority queue Que to implement the pool of jobs.
Note 1 : Implement the priority queue using linked list.
Note 2 : The values of the JID are unique. If there are more than one job with the same p value, then the priority is based on the increasing order of their JID (smaller integer value having higher priority).

    - Your program should be implemented using the following functions:
    - Main() – Repeatedly reads an input character from the menu list through the terminal and executes menu driven operations accordingly. The menu list is ['a', 's', 'n', 'r', 'd' ,'e']. The program ends on input 'e'.
    - ADD(Que, JID, p) – For input 'a', this function should be executed. It adds a new job with job-id as JID and priority as p, to the pool of jobs.
    - SCHEDULE(Que) – For input 's', this function should be executed. It schedules the job JID with the highest priority p from the pool of jobs for execution and then removes it from the pool of jobs and prints it (NOTE: print "-1" if the pool is empty).
    - NEXT-JOB(Que) – For input 'n' , this function should be executed. It displays the JID of the next job to be executed, which will be the JID with the highest priority p from the pool of jobs. (NOTE: the job is not removed from the pool, print "-1" if the queue is empty).

- REPLACE-PRIORITY(Que, JID, np) – For input 'r' , this function should be executed. It replaces the priority of job JID in the pool of jobs with the new priority, np. (NOTE: print "-1" if the JID is not in the pool of jobs).

- DISPLAY(Que) – For input 'd', this function should be executed. It prints the details (JID and p) of each job in the pool of jobs (NOTE: print "-1" if the pool is empty).

Input format:

- Your input can contain two parts separated by space, the first part is the input character from the menu list. The second part feeds the parameters according to the menu option. Which can have at most two integer values (either nothing or one or two).

- Each line of your input contains a character from the menu list ['a', 's', 'n', 'r', 'd', 'e'], followed by at most two integers depending on the menu function (either nothing, or two). The integer values, if given, should be in the range [1, 100].

- Input 'a' should be followed by two positive integers, representing job-id as JID and priority as p in the order, respectively. A new node containing this JID and p is appended to the pool of jobs by calling the function ADD(Que, JID, p) .

- Input 's' calls the function SCHEDULE(Que) which prints and removes the JID of the job with the highest priority p from the pool of jobs.

- Input 'n' calls the function NEXT-JOB(Que) which prints the JID of the job with the highest priority p from the pool of jobs, without removing it from the queue.

- Input 'r' should be followed by two positive integers, representing job-id as JID and a new priority as np in order, respectively. It calls the function REPLACE-PRIORITY(Que, JID, np) which replaces the priority value p of the job JID in the pool of jobs, with the new priority, np.

- Input 'd' calls the function DISPLAY(Que). It prints the details JID p (space separated) of every job in the pool of jobs Que . Each pair of (JID p) should be printed in a new line.

- Input 'e' terminates the execution of the program.

Output format:

- Line can contain an integer or two integers representing JID and priority, respectively.

- Line could also contain the integer "-1" depending upon input.

Sample Input :
```
a 1 45
a 6 46
a 4 32
a 2 34
s
n
d
r 4 36
d
e
```

Sample Output:
```
6
1
1 45
2 34
4 32
```

1 45
4 36
2 34