National Institute of Technology Calicut
Department of Computer Science and Engineering
Third Semester B. Tech.(CSE)
CS2092D Programming Laboratory
Assignment #5 Practice (25-09-2023)

Naming Conventions for Submission

- The source codes must be named as

`ASSG<NUMBER>_<ROLLNO>_<FIRST-NAME>_PRACTICE.c`

(For example: $ASSG5\_BxxyyyyCS\_LAXMAN\_PRACTICE.c$).

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.
- **No need to create a folder or a zip file; the source code itself is sufficient for uploading with the naming convention mentioned above.**
- Global and/or static variables should not be used in your program.

QUESTION

1. Write a menu driven program to implement the hospital doctor-consultation management system. Each patient arriving at the hospital has a booking-number BN ($1 \leq \text{BN} \leq 500$) and a patient-id PID ($1 \leq \text{PID} \leq 10^4$). Patients may keep arriving. The patients are added to a waiting list as they arrive based on their booking-number and are allowed to consult the doctor when their booking number is called. The patients are removed from the list when they are called for consultation. Patients with PID, ($1 \leq \text{PID} \leq 10^3$) are assigned to doctor A and ($1001 \leq \text{PID} \leq 10^4$)are assigned to doctor B. No two patients in the waiting list have the same booking number and same patient id. The patients in the front of the waiting list will be the first to be called for consultation. Use singly linked-list to implement this system.

    - MAIN() - repeatedly reads a character '*a*', '*c*', '*n*', '*bn*', '*pid*', '*dA*', '*dB*', '*d*' or '*e*' from the terminal and calls the sub-functions appropriately until character '*e*' is entered.
    - ADD-PATIENT($LIST, PID, BN$) - adds a new patient with patient-id PID and booking number as BN to the waiting list $LIST$. The patients are placed in the waiting list based on their booking number BN (ascending order of booking number BN)
    - CALL-PATIENT($LIST$) - prints and removes the PID of the patient in the front of the waiting list $LIST$; prints *0* if the list is empty.
    - NEXT-TOKEN($LIST$) - prints the PID of the patient in the front of the waiting list, without removing it from the list; prints *0* if the list is empty.

- GET-BN(*LIST, PID*) - prints the booking number *BN* of the patient with patient id as *PID*, prints *0* if the *PID* is not present in the list.
- GET-PID(*LIST, BN*) – prints the patient id *PID* of the patient with booking number *BN*, prints *0* if the *BN* is not present in the list.
- DOCTORA-PID(*LIST*) - prints the *PID* of patients(space seperated) consulting Doctor A. Prints *0* if no patient is consulting doctor A.
- DOCTORB-PID(*LIST*) - print the *PID* of patients (space seperated) consulting Doctor B. Prints *0* if no patient is consulting doctor B.
- DISPLAY(*LIST*) – prints the details *PID BN* (space seperated) of each patient in the waiting list *LIST* each pair of *PID BN* is printed in new line; prints *0* if the list is empty.

Input format:

- Each line contains a character from '*a*', '*c*', '*n*', '*bn*', '*pid*', '*dA*', '*dB*', '*d*' or '*e*' followed by zero, one or two integers. The integers, if given, are in the range $[1, 10^4]$.
- Character '*a*' is followed by two positive integers, representing patient id 'PID' and booking number 'BN' . A new node containing above details need to be appended to the linked list $LIST$, calling the function ADD-PATIENT($LIST, PID, BN$).
- Character '*c*' is to call the function CALL-PATIENT($LIST$) .
- Character '*n*' is to call the function NEXT-TOKEN($LIST$).
- Character '*bn*' is is followed by one positive integer, representing patient id 'PID' . It calls the function GET-BN(*LIST, PID*) and prints the booking number *BN* of the patient with patient id as *PID*, prints *0* if the *PID* is not present in the list.
- Character '*pid*' is is followed by one positive integer, representing booking number 'BN' . It calls the function GET-PID(*LIST, BN*) and prints the patient id *PID* of the patient with booking number as *BN*, prints *0* if the *BN* is not present in the list.
- Character '*dA*' is to call the function DOCTORA-PID($LIST$)
- Character '*dB*' is to call the function DOCTORB-PID($LIST$)
- Character '*d*' is to call the function DISPLAY($LIST$)
- Character '*e*' is to 'exit' from the program.

Sample Input :

```
a 1011 10
a 5011 5
a 100 20
c
a 151 7
a 1362 21
n
dA
dB
bn 151
pid 21
d
e
```

Sample Output:

```
1011
5011
151 100
5011 1362
7
```

1362
5011 5
151 7
100 20
1362 21