An-Najah Nation University		جامعة النجاح الوطنية
Faculty of Engineering and Information Tech.		كلية الهندسة وتكنولوجيا المعلومات

Computer Engineering Department
Data Structures and Algorithms (10636211)

HW 2

ILOs [iii]	**Due to: 09/11/2019**	**20 points**

Time Scheduling is the process by which tasks are assigned to machines that execute the required tasks. In this homework, you have to write a program that reads information about a set of concurrent tasks from a file and finds the minimum number of machines to execute these tasks.

You have to store the following information about each task:

- Name: Task Name (string)

- ST: Starting Time (int)

- ET: Ending Time (int)

Your program should read the required information about the tasks from a file in the following format:

- The first line gives the number of tasks (N)

- N lines, where each line gives the task-name, start-time, and end-time

```
20
Task_A    1    5
Task_B    2    7
Task_C    6    10
Task_D    1    8
.
.
```

The result of your scheduling program is a linked list of machines, where each machine has a linked list of tasks that can be executed without overlapping.

1

You can use the following definitions:

```
struct TASK {
        char TaskName[32];
        int ST;
        int ET;
        TASK* nextTask;
};

struct MACHINE {
        int id;
        TASK* firstTask;
        MACHINE* nextMachine

};
```
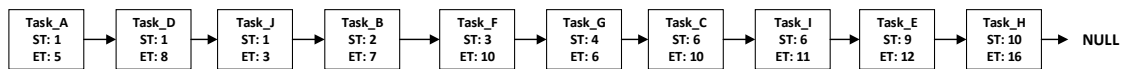
Your program should work as follows:

1) Ask the user to enter the name of the input file.

2) Open the file and read the required information

3) Construct a general linked list of tasks to store the information about the different tasks (The tasks in the lists should be sorted according to their start times).

4) Create an empty list of machines.

5) While the general list of tasks is not empty, do the following:

   a) Remove the first node from the general list.

   b) Try to assign the task to one of the available machines without overlapping.

   c) If you could not find a free machine to execute the task with specified time slot, a new machine should be allocated and added to the list of machines.

6) For each machine, print the assigned tasks as well as the starting and ending time for each task.

Example:

Input File:

```
10
Task_A        1       5
Task_B        2       7
Task_C        6       10
Task_D        1       8
Task_E        9       12
Task_F        3       10
Task_G        4       6
Task_H        10      16
Task_I        6       11
Task_J        1       3
```
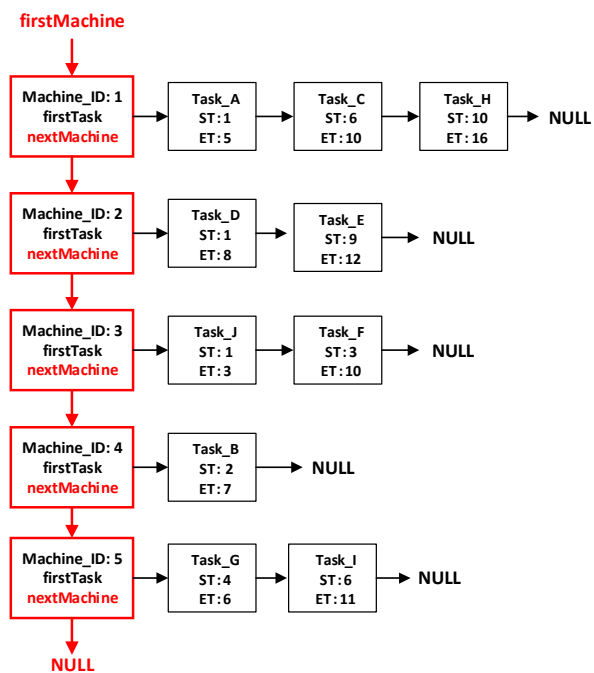
The General Sorted Linked List:

| Task_A ST:1 ET:5 | → | Task_D ST:1 ET:8 | → | Task_J ST:1 ET:3 | → | Task_B ST:2 ET:7 | → | Task_F ST:3 ET:10 | → | Task_G ST:4 ET:6 | → | Task_C ST:6 ET:10 | → | Task_I ST:6 ET:11 | → | Task_E ST:9 ET:12 | → | Task_H ST:10 ET:16 | → NULL |

The Result of Scheduling:

firstMachine

| Machine_ID: 1 firstTask nextMachine | → | Task_A ST:1 ET:5 | → | Task_C ST:6 ET:10 | → | Task_H ST:10 ET:16 | → NULL |

| Machine_ID: 2 firstTask nextMachine | → | Task_D ST:1 ET:8 | → | Task_E ST:9 ET:12 | → NULL |

| Machine_ID: 3 firstTask nextMachine | → | Task_J ST:1 ET:3 | → | Task_F ST:3 ET:10 | → NULL |

| Machine_ID: 4 firstTask nextMachine | → | Task_B ST:2 ET:7 | → NULL |

| Machine_ID: 5 firstTask nextMachine | → | Task_G ST:4 ET:6 | → | Task_I ST:6 ET:11 | → NULL |

NULL

```
Output:

Machine # 1: Task_A(1-5) --- Task_C(6-10) --- Task_H(10-16)
Machine # 2: Task_D(1-8) --- Task_E(9-12)
Machine # 3: Task_J(1-3) --- Task_F(3-10)
Machine # 4: Task_B(2-7)
Machine # 5: Task_G(4-6) --- Task_I(6-11)
```