DATA STRUCTURES

BATCH – A
[TUESDAY MARCH 14, 2017: 2:00 PM – 5:00 PM]

ASSIGNMENTS – 9 CODE: assign09

INSTRUCTIONS: [Total Marks: 30]

i) Read all assignments and each problem has to be answered in the same c file.
ii) Create a .c file following the file name convention: abc-assign09.c
Where abc is your roll number and assign09 is the assignment code
iii) Strictly follow the file name convention and do not use scanf()

PROBLEMS: (on BFS / DFS / Heaps / HeapSort)

1) **[Marks: 5 marks]**

a) Define a heap – HEAP – with MOVIE as its elements.
b) Define a movie – MOVIE - a node of the heap with following fields:

```
movieID: <int> - a Unique Movie Identifier - [100, 2000]
mname: <char> - it can hold a maximum of 16 characters
likes: <int> - size [50, 10000] – likes of the people
year: <int> - year of release of the movie
rating: <float> - [1, 10] – rating of the movie
budget: <float> - [10.0, 20.00] – making cost of the movie(in crores)
revenue: <float> - total box office collection made by the movie in
                   the range - [1.0, 1000.0] (amount in crores)
time: <int> – [20.0, 180.0] – running time in minutes
```

The values of these fields could be generated using a random number generator in the specified range. Assume a list of specific names for the field “mname”

2) **[Marks: 25 marks]**
Using above data structure and function prototypes given below, write your code for following tasks:

a) **[Marks: 3 marks]**
Assume that we need to generate a HEAP with n (=20) movies. Create a heap (specifically Max HEAP) with n MOVIEs.

```
HEAP *createMovieDataset(HEAP *heap, int n);
```

This function should internally insert an element of type MOVIE into the Max HEAP in such a way that the node insertion is based on the field “time”.

b) **[Marks: 2 marks]**
Write a function to print the details of each item one per line:

```
void printMovieDetails(HEAP *movie);
```

c) **[Marks: 6 marks]**
Write a function to perform breadth first search and print the details of the movie visited each time along with their levels information.

```
void performBreadthFirstSearch(HEAP *heap);
```

You may write additional functions separately to perform Breadth First Search and to print the details of a MOVIE.

d) **[Marks: 6 marks]**
Write a function to convert the heap into the linked list in which a way that nodes in the linked list are organized in a specific order.

```
void convertHeapToSortedLinkedList(HEAP *heap);
```

You may write additional functions and assume enough space to create a linked list.

e) **[Marks: 8 marks]**

Write a function to perform sorting (using only the Heap Sort) of the movies based in a specific field: `rating`

```
HEAP *sortMovies (HEAP *movie, <data_type> field);
```

Print the details of the movies in sorted order.

The output should display the highly rating movie first and second highly rated movie next and so on.