

DATA STRUCTURES

BATCH – A

[TUESDAY FEBRUARY 21, 2017: 2:00 PM – 5:00 PM]

ASSIGNMENTS – 6

CODE: **assign06**

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-assign06.c**
Where **abc** is your roll number and **assign06** is the assignment code
- iii) Strictly follow the file name convention and do not use **scanf()**

PROBLEMS:

1) **[Marks: 4 marks]**

Define an ITEM using typedef with the following attributes:

```
proID      : <int> (Unique number)
name       : <string> (Max size: 50 Characters)
type       : <int> (0 - digital SLR, 0 - cybershot,
                  2 - webcam, 3 - others cameras)
make       : <string> (Manufacturer / brand name)

mrp        : <float> (two digits precision)
discount   : <float> (possible deduction from the cost)
charges    : <float> (Delivery charges)

starrate   : <float> (Feed back based on star rating)
```

2) **[Marks: __ marks]**

Using the above data structure and the given function prototypes,
Write your code for the following tasks:

a) **[Marks: 6 marks]**

Read data from the given file – **inputdata.txt** - that contains n (=12) items. Now populate the dataset using the linked list. Each line carries the details of an item and the value of an attribute is delimited by a comma (,).

```
ITEM *genDataset(FILE *fp);
```

This function should internally call the following function to insert an element into the dataset.

```
ITEM *insertItem(ITEM *item);
```

The pointer should always point to the first element.

You may have to parse the input and convert (typecast) it into specific data types. You may use **atoi()** and **atof()** functions for such conversions.

b) **[Marks: 3 marks]**

Write a function to print name, type and price of each item:

```
void printItems(ITEM *item);
```

c) **[Marks: 3 marks]**

Write a function to search and print the item by brand name (by make)

```
void SearchItemByMake(ITEM *item, char *make);
```

d) **[Marks: 4 marks]**

Write a function to search and print details of the item with the lowest cost

```
void SearchLowPrice(ITEM *item);
```

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

Write a function to modify the discount of all items by a specific percentage – off and print the items with the updated discount.

```
void *modifyDiscount(ITEM *item, float off);
```

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

Write a function to delete an item by their product ID

```
void deleteItemByID(ITEM *item, int proID);
```

Note that you have to print the details of the item before deleting it.