

**Independent University, Bangladesh**  
**Department of Computer Science & Engineering**  
**CSE 203 - Data Structure**

**1 hour 20 minutes**

**45 marks**

Implement a system for Cineplex that stores the following information of each movie:

1. Name
2. Release year
3. Number of seats available
4. Ticket price

The system can perform the following operations:

1. Add new movie (You can assume that same movie will not be added multiple times)
2. Buy tickets of a movie. To buy tickets of a movie, user will input the name of the movie and number of tickets he wants to purchase. If the number of tickets exceeds the number of seats available, show user the number of seats available. Otherwise, give him the tickets and adjust the number of seats available accordingly.
3. Update the ticket price of a specific movie.

You can assume that more operation 2 and 3 will be performed more frequently than operation 1. In this context, which would be a better data structure for handling these operations (Array or Linked List or Binary Search Tree? Explain briefly within 5-6 sentences. Next, use the appropriate data structure to implement this system.

N.B:

- a. No need to write down the main function.
- b. Use meaning names for variables and functions.
- c. The use of STL is forbidden.

15

2. Examine the following code and answer the corresponding questions.

- a. Draw the call stack of the code. You may use separate diagrams for calling and returning sequence. Make sure that the diagrams include argument and returning values.
- b. What will be the output of the code?
- c. How many time will the myfunction() be called for the given input ("1230421")?

10

```

1  #include <iostream>
2
3  using namespace std;
4
5  int myfunction(string s, int starting, int ending)
6  {
7      if(ending-starting==1 || starting==ending)
8      {
9          return 1;
10     }
11     if(s[starting]==s[ending])
12     {
13         return myfunction(s,starting+1,ending-1);
14     }
15     return 0;
16 }
17
18 int main()
19 {
20     string s = "1230421";
21
22     cout<<myfunction(s, 0, s.length()-1)<<endl;
23
24     return 0;
25 }

```

3. Suppose, you are given two singly linked list. Write down a program that prints the common elements of the lists without repetition (e.g., the same value will not get printed twice).

You must use the following structure to implement the program.

10

```

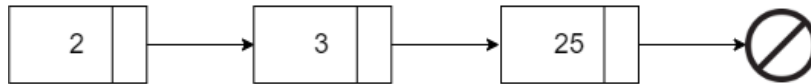
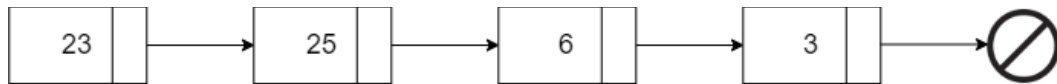
struct node
{
    int value;
    node* next;
};
node* head;

```

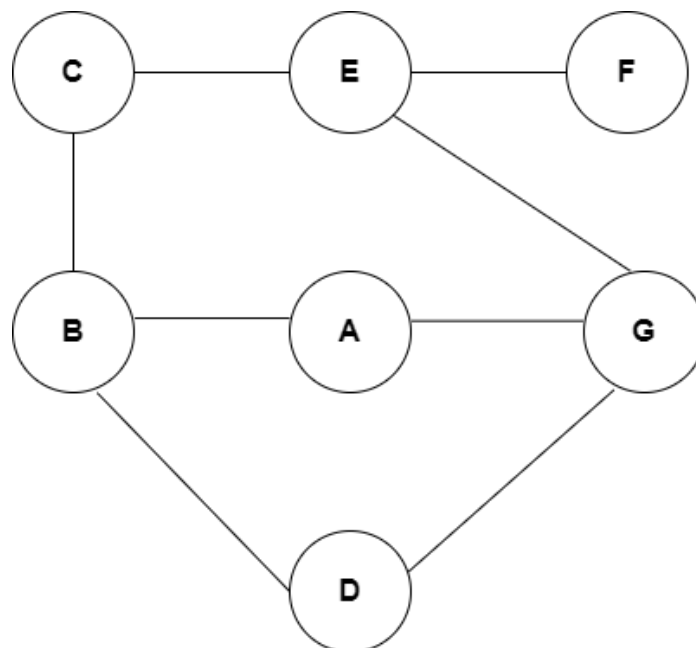
N.B:

- The ordering of values does not matter.
- No need to write down the main function.
- Use meaning names for variables and functions.
- The use of STL is forbidden.

For the following 2 lists, the function will print 25 and 3.



4. Execute BFS and DFS on the following graph. Suppose, the starting node is C and its level is 0. 10



For BFS, prepare the following table:

[illegible]

For DFS, prepare the following table:

Node	Starting time	Finishing time	Parent

**\*\*Grades are earned not given.\*\***