

Q.No.1)

```
C++ main.cpp Format Run
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4 #include <ctime>
5
6 class Card {
7 public:
8     static std::vector<std::string> faces;
9     static std::vector<std::string> suits;
10
11     Card(int face, int suit) : face(face), suit(suit) {}
12
13     std::string toString() {
14         return faces[face] + " of " + suits[suit];
15     }
16
17 private:
18     int face;
19     int suit;
20 };
21
22 std::vector<std::string> Card::faces = {"Ace", "2", "3", "4",
    "5", "6", "7", "8", "9", "10", "Jack", "Queen", "King"};
23 std::vector<std::string> Card::suits = {"Hearts", "Diamonds",
    "Clubs", "Spades"};
24
25 class DeckOfCards {
26 public:
```

Q.No.2)

```

1  #include <iostream>
2  #include <vector>
3
4  class IntegerSet {
5  private:
6      std::vector<bool> elements; // Vector of bools to represent
    the set
7
8  public:
9      // Default constructor that initializes the set to be empty
10     IntegerSet() : elements(101, false) {} // 101 elements for 0
    through 100
11
12     // Constructor that initializes the set with an array of
    integers
13     IntegerSet(const int arr[], int size) : elements(101, false)
    {
14         for (int i = 0; i < size; i++) {
15             if (arr[i] >= 0 && arr[i] <= 100) {
16                 elements[arr[i]] = true;
17             }
18         }
19     }
20
21     // Insert an element into the set
22     void insertElement(int element) {
23         if (element >= 0 && element <= 100) {

```