Q.1 Write a program using arrays to display the output in the following form:

```
4 3 2 1

3 2 1

2 1

1
```

Q.2 (Airline Reservations System)

A small airline has just purchased a computer for its new automated reservations system. You've been asked to develop the new system. You're to write an application to assign seats on each flight of the airline's only plane (capacity: 10 seats).

Your application should display the following alternatives: Please type 1 for First Class and Please type 2 for Economy. If the user types 1, your application should assign a seat in the firstclass section (seats 1–5). If the user types 2, your application should assign a seat in the economy section (seats 6–10). Your application should then display a boarding pass indicating the person's seat number and whether it's in the first-class or economy section of the plane.

Use a one-dimensional array of primitive type boolean to represent the seating chart of the plane. Initialize all the elements of the array to false to indicate that all the seats are empty. As each seat is assigned, set the corresponding element of the array to true to indicate that the seat is no longer available.

Your application should never assign a seat that has already been assigned. When the economy section is full, your application should ask the person if it's acceptable to be placed in the first-class section (and vice versa). If yes, make the appropriate seat assignment. If no, display the message "Next flight leaves in 3 hours."

Q.3 (Total Sales)

Use a two-dimensional array to solve the following problem: A company has four salespeople (1 to 4) who sell five different products (1 to 5).

Once a day, each salesperson passes in a slip for each type of product sold.

Each slip contains the following:
a) The salesperson number
b) The product number
c) The total dollar value of that product sold that day

Thus, each salesperson passes in between 0 and 5 sales slips per day. Assume that the information from all the slips for last month is available.
Write an application that will read all this information for last month's sales and summarize the total sales by salesperson and by product. All totals should be stored in the two-dimensional array sales. After processing all the information for last month, display the results in tabular format, with each column representing a salesperson and each row representing a particular product. Cross-total each row to get the total sales of each product for last month.
Cross-total each column to get the total sales by salesperson for last month. Your output should include these cross-totals to the right of the totaled rows and to the bottom of the totaled columns.

Q.4 Write a class to simulate a Stack.

Q.5 Write a class to simulate a Queue.

Q.6 Write a class with a set of static methods to perform foreign exchange conversions.

For e.g., it could do conversion from INR to USID

(Indian Rupee to US Dollar).

Q.7 ***(Savings Account Class)***
Create class SavingsAccount.
Use a static variable annualInterestRate to store the annual interest rate for all account holders. Each object of the class contains a private instance variable savingsBalance indicating the amount the saver currently has on deposit.

Provide method calculateMonthlyInterest to calculate the monthly interest by multiplying the savingsBalance by annualInterestRate divided by 12—this interest should be added to savings-Balance. Provide a static method modifyInterestRate that sets the annualInterestRate to a new value. Write a program to test class SavingsAccount. Instantiate two savingsAccount objects, saver1 and saver2, with balances of $2000.00 and $3000.00, respectively. Set annualInterestRate to 4%, then calculate the monthly interest for each of 12 months and print the new

balances for both savers. Next, set the annualInterestRate to 5%, calculate the next month's interest and print the new balances for both savers.

Q.8 (Rational Numbers)

Create a class called Rational for performing arithmetic with fractions.
Write a program to test your class.
Use integer variables to represent the private instance variables of the class—the numerator and the denominator.
Provide a constructor that enables an object of this class to be initialized when it's declared. The constructor should store the fraction in reduced form.
The fraction 2/4 is equivalent to 1/2 and would be stored in the object as 1 in the numerator and 2 in the denominator.

Provide a no-argument constructor with default values in case no initializers are provided. Provide public methods that perform each of the following operations:

a) Add two Rational numbers: The result of the addition should be stored in reduced
form. Implement this as a static method.

b) Subtract two Rational numbers: The result of the subtraction should be stored in reduced form. Implement this as a static method.

c) Multiply two Rational numbers:
The result of the multiplication should be stored in reduced form. Implement this as a static method.

d) Divide two Rational numbers: The result of the division should be stored in reduced form.
Implement this as a static method.

e) Return a String representation of a Rational number in the form a/b, where a is the numerator and b is the denominator.
f) Return a String representation of a Rational number in floating-point format. (Consider providing formatting capabilities that enable the user of the class to specify the number of digits of precision to the right of the decimal point.)

Q.9 Write an enum type TrafficLight, whose constants (RED, GREEN, YELLOW) take one parameter—the duration of the light.Write a program to test the TrafficLight enum so that it displays the enum constants and their durations.