1. Write a program in Java that will read up to ten letters into an array and write the letters back to the screen in the reverse order. For example, if the input is `abcd.' then the output should be `dcba'. Use a period as a sentinel value to mark the end of the input.

2. Write a program in Java to implement linear search.

3. Write a program in Java to implement binary search.

3. Write a program in Java to implement bubble sort.

4. Write a program in Java to implement insertion sort.

5. Write a program in Java to fill the array A (declared below) with numbers typed in at the keyboard. The numbers will be input five per line, on four lines (although your solution need not depend on how the input numbers are divided into lines).
        int A[4][5];

6. Write a program in Java to implement matrix multiplication.

7. Write a program in Java to implement merge sort.

8. Write a program that scores a blackjack hand. In blackjack, a player receives from two to five cards. The cards 2 through 10 are scored as 2 through 10 points each. The face cards—jack, queen, and king—are scored as 10 points. The goal is to come as close to a score of 21 as possible without going over 21. Hence, any score over 21 is called "busted." The ace can count as either 1 or 11, whichever is better for the user. For example, an ace and a 10 can be scored as either 11 or 21. Since 21 is a better score, this hand is scored as 21. An ace and two 8s can be scored as either 17 or 27. Since 27 is a "busted" score, this hand is scored as 17.

The user is asked how many cards she or he has, and the user responds with one of the integers 2 , 3 , 4 , or 5 . The user is then asked for the card values. Card values are 2 through 10, jack, queen, king, and ace. A good way to handle input is to use the type char so that the card input 2, for example, is read as the character '2' , rather than as the number 2 . Input the values 2 through 9 as the characters '2' through '9' . Input the values 10, jack, queen, king, and ace as the characters 't' , 'j' , 'q' , 'k' , and 'a' . (Of course, the user does not type in the single quotes.) Be sure to allow upper- as well as lowercase letters as input.

After reading in the values, the program should convert them from character values to numeric card scores, taking special care for aces. The output is either a number between 2 and 21 (inclusive)

or the word Busted.You are likely to have one or more long multiway branches that uses a switch statement or nested if-else statement. Your program should include a loop that lets the user repeat this calculation until the user says she or he is done.

9. The following problem is sometimes called "The Monty Hall Game Show Problem." You are a contestant on a game show and have won a shot at the grand prize. Before you are three closed doors. Behind one door is a brand new car. Behind the other two doors are consolation prizes. The location of the prizes is randomly selected. The game show host asks you to select a door, and you pick one. However, before revealing the contents behind your door, the game show host reveals one of the other doors with a consolation prize. At this point, the game show host asks if you would like to stick with your original choice or switch your choice to the other closed door. What choice should you make to optimize your chances of winning the car? Does it matter whether you stick with your original choice or switch doors?

Write a simulation program to solve the game show problem. Your program should make 10,000 simulated runs through the problem, randomly selecting locations for the prize, and then counting the number times the car was won when sticking with the original choice, and counting the number of times the car was won when switching doors. Output the estimated probability of winning for both strategies. Be sure that your program exactly simulates the process of selecting the door, revealing one, and then switching. Do not make assumptions about the actual solution (for example, simply assuming that there is a 1/3 or 1/2 chance of getting the prize).

10. You run four computer labs. Each lab contains computer stations that are numbered as shown in the table below:

Lab Number	Computer Station Numbers
1	1-5
2	1-6
3	1-4
4	1-3

Each user has a unique five-digit ID number. Whenever a user logs on, the user's ID, lab number, and the computer station number are transmitted to your system. For example, if user 49193 logs onto station 2 in lab 3, then your system receives (49193, 2, 3) as input data. Similarly, when a user logs off a station, then your system receives the lab number and computer station number. Write a computer program that could be used to track, by lab, which user is logged onto which computer. For example, if user 49193 is logged into station 2 in lab 3 and user 99577 is logged

into station 1 of lab 4 then your system might display the following:

```
Lab Number Computer Stations
1          1: empty 2: empty 3: empty 4: empty 5: empty
2          1: empty 2: empty 3: empty 4: empty 5: empty 6: empty
3          1: empty 2: 49193 3: empty 4: empty
4          1: 99577 2: empty 3: empty
```

Create a menu that allows the administrator to simulate the transmission of information by manually typing in the login or logoff data. Whenever someone logs in or out, the display should be updated. Also write a search option so that the administrator can type in a user ID and the system will output what lab and station number that user is logged into, or "None" if the user ID is not logged into any computer station.