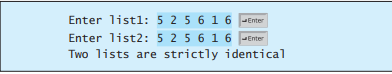
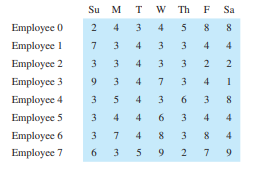
**PROGRAMS FOR PRACTICE**

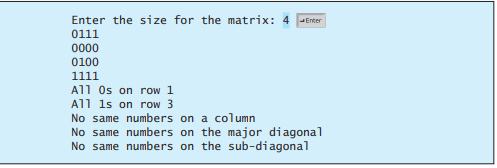
1. Random shuffling the elements in an array
2. Write a program that finds the number of items above the average of all items.
3. Write a program to count the occurrences of each letter in an array of characters.
4. Searching
5. Write a program that prompts the user to enter the number of students, the students’ names, and their scores, and prints student names in decreasing order of their scores.
6. Write two overloaded methods that return the average of an array with the following headers: public static int average(int[] array) public static double average(double[] array) Write a test program that prompts the user to enter ten double values, invokes this method, and displays the average value.
7. Write a method that finds the smallest element in an array of double values using the following header: public static double min(double[] array) Write a test program that prompts the user to enter ten numbers, invokes this method to return the minimum value, and displays the minimum value
8. Write a method that returns a new array by eliminating the duplicate values in the array using the following method header: public static int[] eliminateDuplicates(int[] list) Write a test program that reads in ten integers, invokes the method, and displays the result.
9. Write a program that randomly generates an array of 100,000 integers and a key. Estimate the execution time of invoking the linearSearch method . Sort the array and estimate the execution time of invoking the binarySearch method. You can use the following code template to obtain the execution time: long startTime = System.currentTimeMillis(); perform the task; long endTime = System.currentTimeMillis(); long executionTime = endTime - startTime;
10. The arrays list1 and list2 are strictly identical if their corresponding elements are equal. Write a method that returns true if list1 and list2 are strictly identical, using the following header: public static boolean equals(int[] list1, int[] list2) Write a test program that prompts the user to enter two lists of integers and displays whether the two are strictly identical. Here are the sample runs.



Suppose the weekly hours for all employees are stored in a two-dimensional array. Each row records an employee’s seven-day work hours with seven columns. For example, the following array stores the work hours for eight employees. Write a program that displays employees and their total hours in decreasing order of the total hours.



1. Write a program that prompts the user to enter the length of a square matrix, randomly fills in 0s and 1s into the matrix, prints the matrix, and finds the rows, columns, and diagonals with all 0s or 1s. Here is a sample run of the program:



1. Design a class named Fan to represent a fan. The class contains:

■ Three constants named SLOW, MEDIUM, and FAST with the values 1, 2, and 3 to denote the fan speed.

■ A private int data field named speed that specifies the speed of the fan (the default is SLOW).

■ A private boolean data field named on that specifies whether the fan is on (the default is false).

■ A private double data field named radius that specifies the radius of the fan (the default is 5).

■ A string data field named color that specifies the color of the fan (the default is blue).

■ The accessor and mutator methods for all four data fields.

■ A no-arg constructor that creates a default fan.

■ A method named toString() that returns a string description for the fan.

If the fan is on, the method returns the fan speed, color, and radius in one combined string. If the fan is not on, the method returns the fan color and radius along with the string “fan is off” in one combined string. Write a test program that creates two Fan objects. Assign maximum speed, radius 10, color yellow, and turn it on to the first object. Assign medium speed, radius 5, color blue, and turn it off to the second object. Display the objects by invoking their toString method.

1. Design a class named Account that contains: ■ A private int data field named id for the account (default 0).

■ A private double data field named balance for the account (default 0).

■ A private double data field named annualInterestRate that stores the current interest rate (default 0). Assume all accounts have the same interest rate.

■ A private Date data field named dateCreated that stores the date when the account was created.

■ A no-arg constructor that creates a default account.

■ A constructor that creates an account with the specified id and initial balance.

■ The accessor and mutator methods for id, balance, and annualInterestRate.

■ The accessor method for dateCreated.

■ A method named getMonthlyInterestRate() that returns the monthly interest rate.

■ A method named getMonthlyInterest() that returns the monthly interest.

■ A method named withdraw that withdraws a specified amount from the account.

■ A method named deposit that deposits a specified amount to the account.

(Hint: The method getMonthlyInterest() is to return monthly interest, not the interest rate. Monthly interest is balance \* monthlyInterestRate. monthlyInterestRate is annualInterestRate / 12. Note that annualInterestRate is a percentage, e.g., like 4.5%. You need to divide it by 100.)

Write a test program that creates an Account object with an account ID of 1122, a balance of $20,000, and an annual interest rate of 4.5%. Use the withdraw method to withdraw $2,500, use the deposit method to deposit $3,000, and print the balance, the monthly interest, and the date when this account was created.

1. The split method in the String class returns an array of strings consisting of the substrings split by the delimiters. However, the delimiters are not returned. Implement the following new method that returns an array of strings consisting of the substrings split by the matching delimiters, including the matching delimiters. public static String[] split(String s, String regex) For example, split("ab#12#453", "#") returns ab, #, 12, #, 453 in an array of String, and split("a?b?gf#e", "[?#]") returns a, b, ?, b, gf, #, and e in an array of String.
2. Design a class named Person and its two subclasses named Student and Employee. Make Faculty and Staff subclasses of Employee. A person has a name, address, phone number, and email address. A student has a class status (freshman, sophomore, junior, or senior). Define the status as a constant. An employee has an office, salary, and date hired. Use the MyDate class defined in Programming Exercise 10.14 to create an object for date hired. A faculty member has office hours and a rank. A staff member has a title. Override the toString method in each class to display the class name and the person’s name. Write a test program that creates a Person, Student, Employee, Faculty, and Staff, and invokes their toString() methods.

17. Design a class named Triangle that extends GeometricObject. The class contains:

■ Three double data fields named side1, side2, and side3 with default values 1.0 to denote three sides of the triangle.

■ A no-arg constructor that creates a default triangle.

■ A constructor that creates a triangle with the specified side1, side2, and side3.

■ The accessor methods for all three data fields.

■ A method named getArea() that returns the area of this triangle.

■ A method named getPerimeter() that returns the perimeter of this triangle.

■ A method named toString() that returns a string description for the triangle.

The toString() method is implemented as follows: return "Triangle: side1 = " + side1 + " side2 = " + side2 + " side3 = " + side3; Draw the UML diagrams for the classes Triangle and GeometricObject and implement the classes. Write a test program that prompts the user to enter three sides of the triangle, a color, and a Boolean value to indicate whether the triangle is filled. The program should create a Triangle object with these sides and set the color and filled properties using the input. The program should display the area, perimeter, color, and true or false to indicate whether it is filled or not.

Write the following method that sorts an ArrayList of numbers: public static void sort(ArrayList list) Write a test program that prompts the user to enter 5 numbers, stores them in an array list, and displays them in increasing order.

1. Write a program that meets the following requirements:

■ Creates an array with 100 randomly chosen integers.

■ Prompts the user to enter the index of the array, then displays the corresponding element value. If the specified index is out of bounds, display the message Out of Bounds.

1. Write a program that removes all the occurrences of a specified string from a text file.
2. Write a program that will count the number of characters, words, and lines in a file. Words are separated by whitespace characters.
3. Suppose that a text file contains an unspecified number of scores separated by blanks. Write a program that prompts the user to enter the file, reads the scores from the file, and displays their total and average.
4. Write a program to create a file named Exercise12\_15.txt if it does not exist. Write 100 integers created randomly into the file using text I/O. Integers are separated by spaces in the file. Read the data back from the file and display the data in increasing order.
5. Design a new Triangle class that extends the abstract GeometricObject class. Draw the UML diagram for the classes Triangle and GeometricObject and then implement the Triangle class. Write a test program that prompts the user to enter three sides of the triangle, a color, and a Boolean value to indicate whether the triangle is filled. The program should create a Triangle object with these sides and set the color and filled properties using the input. The program should display the area, perimeter, color, and true or false to indicate whether it is filled or not.