CA3511 - JAVA PROGRAMMING AND NETWORKS LABORATORY

Lab.Ex.2 02.09.2020

1. ***(Sales Commissions)*** Use a one-dimensional array to solve the following problem: A company pays its salespeople on a commission basis. The salespeople receive $200 per week plus 9% of their gross sales for that week. For example, a salesperson who grosses $5,000 in sales in a week receives $200 plus 9% of $5,000, or a total of $650. Write an application (using an array of counters) that determines how many of the salespeople earned salaries in each of the following ranges (assume that each salesperson’s salary is truncated to an integer amount):
   1. $200–299
   2. $300–399
   3. $400–499
   4. $500–599
   5. $600–699
   6. $700–799
   7. $800–899
   8. $900–999
   9. $1,000 and over

Summarize the results in tabular format.

1. Write statements that perform the following one-dimensional-array operations:
   1. Set the 10 elements of integer array counts to zero.
   2. Add one to each of the 15 elements of integer array bonus.
   3. Display the five values of integer array bestScores in column format.
2. ***(Duplicate Elimination)*** Use a one-dimensional array to solve the following problem:

Write an application that inputs five numbers, each between 10 and 100, inclusive. As each number is read, display it only if it’s not a duplicate of a number already read.

Provide for the “worst case,” in which all five numbers are different.

Use the smallest possible array to solve this problem.

Display the complete set of unique values input after the user enters each new value.

1. Label the elements of three-by-five two-dimensional array sales to indicate the order in

which they’re set to zero by the following program segment:

for (int row = 0; row < sales.length; row++)

{

for (int col = 0; col < sales[row].length; col++)

{

sales[row][col] = 0;

}

}

1. ***(Variable-Length Argument List)*** Write an application that calculates the product of a series

of integers that are passed to method product using a variable-length argument list. Test your method with several calls, each with a different number of arguments.

1. ***(Using the Enhanced for Statement)*** Write an application that uses an enhanced for statement to sum the double values passed by the command-line arguments. [*Hint:* Use the static method parseDouble of class Double to convert a int to a double value.]
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."**

1. ***(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:
   1. The salesperson number
   2. The product number
   3. 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 total rows and to the bottom of the total columns.