ECET11 – Object Oriented Programming LAB activity 3

Objective: vectors

Source: Deitel & Deitel Edition 5 – Chapter 7 Exercises

1. (Salespeople commission) Use a vector to solve the following problem. A company pays its salespeople on a commission basis. The salespeople each receive $200 per week plus 9 percent of their gross sales for that week. For example, a salesperson who grosses $5000 in sales in a week receives $200 plus 9 percent of $5000, or a total of $650. Write a program (using a vector 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):

   a) $200$299
   b) $300$399
   c) $400$499
   d) $500$599
   e) $600$699
   f) $700$799
   g) $800$899
   h) $900$999
   i) $1000 and over

2. (Valid numbers) Use a vector to solve the following problem. Read in numbers, each of which is between 0 and 100, inclusive up to a sentinel value of your choice. As each number is read, validate it and store it in the vector only if it is not a duplicate of a number already read. After reading all the values, display only the unique values that the user entered.

3. (Rolling two dice) Write a program that simulates the rolling of two dice. The program should use rand to roll the first die and should use rand again to roll the second die. The sum of the two values should then be calculated. Each die can show an integer value from 1 to 6, so the sum of the two values will vary from 2 to 12, with 7 being the most frequent sum and 2 and 12 being the

least frequent sums. The figure below shows the 36 possible combinations of the two dice. Your program should roll the two dice 36,000 times. Use a vector to tally the numbers of times each possible sum appears. Print the results in a tabular format.   Also, determine if the totals are reasonable (i.e., there are six ways to roll a 7, so approximately one-sixth of all the rolls should be 7).

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

4. (Airline reservations system) A small airline has just purchased a newer computer for its new automated reservations system. You have been asked to program the new system. You are to write a program to assign seats on each flight of the airline's only plane (capacity: N seats). N is determined by an airline administrator.

Your program should display the following menu of alternatives:

  + Please type 1 for "First Class"
  + Please type 2 for "Economy".

If the person types 1, your program should assign a seat in the first class section (seats 1 → M). M is determined by an airline administrator whereby M = N/5. If the person types 2, your program should assign a seat in the economy section (seats (N/5 + 1) → N). Your program should print a boarding pass indicating the person's seat number and whether it is in the first class or economy section of the plane.

Use a vector to represent the seating chart of the plane. Initialize all the elements of the array to 0 to indicate that all seats are empty. As each seat is assigned, set the corresponding elements of the array to 1 to indicate that the seat is no longer available.

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

5. (The Sieve of Eratosthenes) A prime integer is any integer that is evenly divisible only by itself and 1. The Sieve of Eratosthenes is a method of finding prime numbers. It operates as follows:

a. Create a vector with all elements initialized to 1 (true). Vector elements with prime subscripts will remain 1. All other vector elements will eventually be set to zero. You will ignore elements 0 and 1 in this exercise.
b. Starting with vector subscript 2, every time a vector element is found whose value is 1, loop through the remainder of the vector and set to zero every element whose subscript is a multiple of the subscript for the element with value 1. For vector subscript 2, all elements beyond 2 in the vector that are multiples of 2 will be set to zero (subscripts 4, 6, 8, 10, etc.); for vector subscript 3, all elements beyond 3 in the vector that are multiples of 3 will be set to zero (subscripts 6, 9, 12, 15, etc.); and so on.

When this process is complete, the vector elements that are still set to one indicate that the subscript is a prime number. These subscripts can then be printed. Write a program that uses a vector of N elements, (i.e. determined by the end user), to determine and print the prime numbers. Ignore element 0 of the array.