IT 106
Introduction to IT Problem Solving Using Computer Programming

Programming Assignment 5/6

Scenario:

Class registration is an important aspect to the PatriotWeb system. When registering for a course, you must know the course reference number (CRN). Based on the CRN, there are a number of details that one can identify, such as a course title. Consider the table below which identifies some CRNs and their associated course title. Each course has a maximum enrollment of 30.

CRN	COURSE TITLE
12451	Introduction to IT Problem Solving Using Computer Programming
15349	Object-Oriented Techniques for IT Problem Solving
18467	Applied IT Programming
16890	Database Fundamentals
13334	Database Programming

Create a program for use by the Registrar (user) to enroll students into these courses. A single enrollment into a course occurs when the user enters a CRN. Until the Registrar (user) has indicated they are finished entering CRNs, continue to prompt the user to enter a CRN. You must validate the CRN, providing an error message and re-prompting the user if an invalid CRN is entered. Keep track of the number of enrollments for each course. Course enrollments for a specific CRN cannot exceed its maximum enrollment, or else an error message will appear and the enrollment will not be processed.

Once the user has indicated they are finished entering CRNs, display a well-formatted report containing a list of each course with its CRN and title, the number of students enrolled and the number of seats remaining.

To Do (Check Blackboard for Due Dates):

Programming Assignment 5: Solution Design

1) Create a defining diagram that shows the input, processing, and output
2) Create a solution algorithm using pseudocode
3) Show testing using the desk checking table method, to include test data, expected results, and a desk checking table. Make sure your desk checking considers multiple cases including both valid and invalid test data to prove your algorithm will work

Upload a Word document containing only items above to Blackboard.

Grading Criteria	
Requirement	**Points**
Defining Diagram with input, processing, and output	40
Efficient Solution Algorithm	40
Thorough Desk Checking Table including test data, and expected results	20

Full points will be awarded for an accurate, efficient, complete defining diagram, solution algorithm, and desk checking table. Partial credit is available.

Programming Assignment 6: Solution Implementation

Write a well-documented, efficient Java program that implements the algorithm you identified. Include appropriate documentation as identified in the documentation expectations document.

Note: You may not use the Scanner or System.out classes. For input/output, you must use the JOptionPane class.

Upload the .java file of the final program to Blackboard.

Full points will be awarded for an accurate, efficient, complete Java program. Partial credit is available. Any final program that does not compile will receive an automatic zero.

Grading Criteria	
Requirement	**Points**
Implementation of Java Program, using efficient practices where appropriate, such as the use of constants, good variable names, no redundant code, etc.	70
Appropriate objective-style documentation	10
Appropriate intermediate comments	20