CS 180 Lab 9 Due by beginning of class on Friday, 10/27/17

This lab investigates enumerated data types, two-dimensional arrays, parallel one-dimensional arrays, the Get\_Line procedure, and I/O of enumerated data types.

Your task is to first, understand the code provided below. Then modify the following code to print the column headings (as enumerated types, not strings) above their respective data in the two-dimensional array.

Each record of the input file (an initial test file is provided on Blackboard) consists of 5 scores: a program score, quiz score, homework score, lab score, test score, and student name, in this order. Of particular note, is the fact that the names are not any specific length and they do not have blanks appended to make them uniform length.

The program below reads all the data, stores it in the appropriate data structures. You need to complete the procedure that outputs the data. One of the things you’ll need to do is figure out how to make the column headings (PROGRAM, QUIZ, HOMEWORK, LAB, TEST) appear above their respective columns as you print the two-dimensional array; note that you must output the enumerated type as the enumerated type, itself and not convert it to a string before output.

WITH Ada.Text\_Io;

USE Ada.Text\_Io;

WITH Ada.Integer\_Text\_Io;

USE Ada.Integer\_Text\_Io;

WITH Ada.Float\_Text\_IO;

USE Ada.Float\_Text\_IO;

PROCEDURE Lab9 IS

TYPE Assignment\_Type IS

(Program,

Quiz,

Homework,

Lab,

Test);

--two-dimensional array type definition

TYPE Score\_Array\_Type IS ARRAY (1 .. 100, Assignment\_Type) OF Natural;

--array of strings

SUBTYPE Name\_Type IS String(1..25);

TYPE Name\_Array\_Type IS ARRAY (1 .. 100) OF Name\_Type;

--what is the purpose of this array type?

TYPE Name\_Length\_Array\_Type IS ARRAY (1 .. 100) OF Natural;

--instantiate the generic package to do I/O of Assignment\_Type literals

--complete the statement below

PACKAGE ASSIGNMENT\_TYPE\_IO IS NEW

USE ASSIGNMENT\_TYPE\_IO;

-------------------------------------------------------------------------------------

PROCEDURE Get\_Names\_And\_Scores (

Scores\_Fp : OUT Score\_Array\_Type;

Names\_Fp : OUT Name\_Array\_Type;

Name\_Lengths\_Fp : OUT Name\_Length\_Array\_Type;

N\_Fp : OUT Natural) IS

BLANK : CHARACTER;

BEGIN

N\_Fp := 0;

WHILE NOT End\_Of\_File LOOP

N\_Fp := N\_Fp + 1;

FOR Col\_INDEX IN ASSIGNMENT\_Type LOOP

Get(SCORES\_FP(N\_FP,COL\_INDEX));

END LOOP;

GET(BLANK);

Get\_Line(

Names\_Fp (N\_Fp),

Name\_Lengths\_Fp (N\_Fp));

END LOOP;

END Get\_Names\_And\_Scores;

--------------------------------------------------------------------------------------

PROCEDURE Print\_Name\_And\_Score\_Arrays (

Names\_Fp : IN Name\_Array\_Type;

Scores\_Fp : IN Score\_Array\_Type;

NAME\_LENGTH : IN NAME\_LENGTH\_ARRAY\_TYPE;

N\_Fp : IN Natural) IS

BEGIN

--complete this procedure so that it outputs each student’s name followed by all their scores

--don’t forget to output, as an enumerated data type, the column heading label (program, quiz, homework, etc)

--above its respective score. Good luck!!

END Print\_Name\_And\_Score\_Arrays;

------------------------------------------------------------------------------------

--main program declaration section

N : Natural;

Scores : Score\_Array\_Type;

Names : Name\_Array\_Type;

Name\_Lengths : Name\_Length\_Array\_Type;

BEGIN --main program

Get\_Names\_And\_Scores(Scores, Names,Name\_Lengths,N);

Print\_Name\_And\_Score\_Arrays(Names,Scores,NAME\_LENGTHS,N);

END Lab9;