**Lab5: Advanced Loops**

**Lab 5.1 – Pseudocode**

In this portion of the lab you will analyze the problem listed below. You will figure out what the inputs and outputs are as well as develop a problem statement. You will also determine what variables you will use and create some pseudocode.

**Alberta Einstein teaches a business class at Podunk University. To evaluate the students in this class, she has given three tests. It is now the end of the semester and Alberta would like to have a program that inputs each student’s name and test scores and outputs the average score for each student and the overall class average. When Ms. Einstein is done inputting all of her student grades, have her enter *no* to a question asking her to end the program. (Hint: The outer loop should allow for Ms. Einstein to input all the students, one by one, and the inner loop should accept the three exam scores, and compute the average for each student.)**

**Step 1**: Analyze the problem. In the table below list the inputs, outputs and a one sentence description of the problem.

|  |  |
| --- | --- |
| Input: | Student name, test score 1, test score 2, test score 3 |
| Problem Statement: | Calculate the average for each student. |
| Output: | Student name, test score 1, test score 2, test score 3, test average |

**Step 2:** What variables are you going to need and what will be their datatype. (The number of spaces in this table does not necessarily mean these are the only variables\constants. This problem can be solved a number of different ways.)

|  |  |
| --- | --- |
| **Variable\Constant Name** | **Datatype** |
| Student name | String |
| Test score 1 | Float |
| Test score 2 | Float |
| Test score 3 | Float |
| Test average | float |

**Step 3:** Create a 30,000 view outline of the problem. How would you break this problem into numerous smaller units? Place your outline below:

|  |
| --- |
| * Declare my variables * Gather inputs * Calculate the average * Output results * Loop till complete |

**Step 4:** In looking at your overall outline of the problem. Break apart your larger portions into smaller ones. Place your revised outline below:

|  |
| --- |
| * Declare my variables   + Declare the name as a string   + Declare the scores as a float * Gather inputs   + Ask for student names   + Ask for student test scores * Calculate the average   + Add all the students scores and divide by the number of tests (3) * Output results   + Output the students name, test scores, and average * Loop till complete   + Ask if she wishes to continue   + If no end the program |
|  |
|  |
|  |

**Lab5.2 – Raptor**

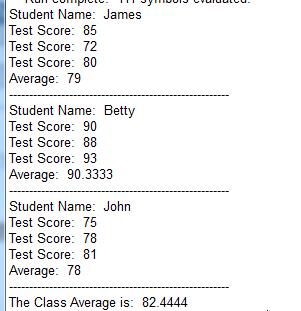
In this portion of the lab you will take the pseudocode you developed above and test it using Raptor.

**Step 1:** Code in Raptor. Open up Raptor and code this problem.

**Step 2**: Test your program using the following data. At this point don’t worry about getting your prices to display only 2 decimal places. We will learn how to do that later.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student Name | Test Score #1 | Test Score #2 | Test Score #3 | Average |
| James | 85 | 72 | 80 | 79 |
| Betty | 90 | 88 | 93 | 90.3333 |
| John | 75 | 78 | 81 | 78 |
| Class Average |  |  |  | **82.4444** |

My output looked like the following:



**Step 3**: If your results match the test, then hand in your Raptor program and this lab. Otherwise go back over your Raptor program and correct any errors.

**Lab5.3 – Python**

In this portion of the lab you will take the Raptor file you just created and translate it to a Python file. You will also gain more experience utilizing Python’s formatting characters.

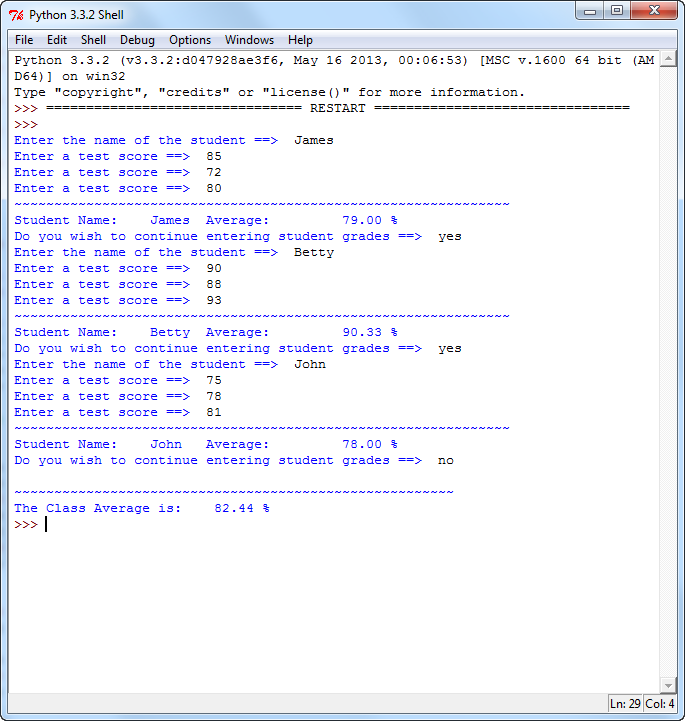
**Step 1:** Open up your Raptor file and resize the window so that it takes up half of your computer screen.

**Step 2:** Open IDLE (Python compiler). Click on File and select New Window. In the new window click on Save. Give your file a name. Resize it so that it takes up half of your screen just like you did previously in this lab.

**Step 3:** Translate your Raptor file into Python code. Remember in Raptor we cannot utilize WHILE or FOR loops. However, we have both syntaxes available to us in Python. Also don’t forget to utilize the format and tab characters for your output. Round your averages to 2 decimal places.

**Step 4:** Test and Debug your Python code using the following test data.

My output looked like the following:



**Step 5**: If your results match the test, then you are done. Otherwise go back over your code and correct any errors.

**How Will I Be Graded**

You will be graded based off of the following rubric

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exemplary** | **80 %** | **Competent** | **60%** | **Developing** |
| 3 | Completed the analysis of inputs and outputs, developed a correct problem statement and developed thorough pseudocode. | 8 | Did not complete the input and output analysis or the problem statement was not detailed enough. | 6.0 | Did not develop the analysis of the inputs and outputs and problem statement or the pseudocode was lacking detail. |
| 3.5 | Correctly coded the problem in Raptor. Includes comments, variable declarations, input statements, process statements and correct output | 2.8 | Coded problem in Raptor but a variable declaration, an input statement or an output statement. | 2.1 | Completed the problem in Raptor but there was an error in the processing |
| 3.5 | Correctly coded the problem in Python. Includes variable declarations, comments, input statements, correct utilization of while loop and for loop syntax, math is correct and output has utilized the tab and formatting commands. | 2.8 | Missing one item from Exemplary | 2.1 | Missing more than one item in Exemplary |
| 10 | TOTAL POINTS | 10.8 |  | 8.1 |  |

**Handing it all in**

1. Log on to Canvas and our course shell.
2. Upload this lab sheet along with your Raptor and Python files to the Lab #5 drop box.