UNIVERSITY OF CALOOCAN CITY
Caloocan, 1400 Metro Manila, Philippines

COLLEGE OF ENGINEERING
Computer Engineering
2nd Semester, School Year 2024-2025

Laboratory Activity No. 2.2	
Literals, Operators, Variable, and Supplementary Activity	
Course Code: CPE103	**Program:** BSCPE
Course Title: Object-Oriented Programming	**Date Performed:** February 1, 2025
Section: 1- A	**Date Submitted:** February 8, 2025
Name: Directo, Hannah Thea B.	**Instructor:** Engr. Maria Rizette Sayo

1. Objective(s):

1. Implement literals and variables in a python program.
2. Learn how to calclulate grade using the python program.

2. Intended Learning Outcomes (ILOs):

At the end of this activity, students should be able to:

1. Write simple program implementing literals and variables.
2. Use comments and identify keywords from identifiers created by users.

3. Discussion

In Python, **variables** are used to store data, acting as named references to memory locations where values can be assigned and modified. These values can be of various data types, such as integers, strings, floats, or lists.

Constants are variables whose values should not change during the program's execution. Although Python does not have built-in support for constants, you can use a naming convention to indicate that a variable should be treated as a constant.

Literals are notations for representing fixed values in source code. They are constants that are self-explanatory and do not need to be computed or evaluated. Python supports various types of literals, such as numeric literals, string literals, Boolean literals, and more.

4. Materials and Equipment:

Desktop Computer with Python Colab
Windows Operating System

5. Procedure

1. I analyzed the given task and ordered the needed for grades to be inputted then proceeded to do the basic float and input function for the user to input the grades.
2. Then, I did the code line for the computation of the class standing including the assignment, quiz and hands-on activities and the exam grade in order to get the Midterm grade, Prelim grade and the Final Grade.
3. I used the spacing "\n" in order to add spacing for the summary of grades, where it printed the name and the grades of the student.

6. Supplementary Activity:

Task:

1. A teacher wants to calculate the final grade in a CpE course and want to write it in a python program. The following are the requirements:

1. PRELIM GRADE = 50% Prelim Exam + 50% Prelim Class Standing (CS)
2. PRELIM CS = 50% Hands-on activity + 30% Quiz + 20% Assignment
3. MIDTERM GRADE = 1/3 of PRELIM GRADE + 2/3 of (50% Midterm Exam + 50% Midterm Class Standing (CS))
4. MIDTERM CS = 50% Hands-on activity + 30% Quiz + 20% Assignment
5. FINAL GRADE = 1/3 of MIDTERM GRADE + 2/3 of (50% Final Exam + 50% Final Class Standing (CS))
6. FINAL CS = 50% Hands-on activity + 30% Quiz + 20% Assignment
7. HOAs, Quizzes and Assignments are inputted as average of all submissions and are out of 100%.
8. Major exams are inputted out of 100%.
9. Show the codes that successfully run the program.
10. Provide comments or documentation strings for your program.

Supplementary Activity:

1. Test 3 students from the program you created.
2. The program should show the name of the student, the PRELIM, MIDTERM and FINAL grades.
3. Convert the final grade into the UCCs numerical grade. Please refer to the grading system.

Refer to this link:

Activity_2_2.ipynb - Colab

Questions:

UNIVERSITY OF CALOOCAN CITY
Caloocan, 1400 Metro Manila, Philippines

COLLEGE OF ENGINEERING
Computer Engineering
2nd Semester, School Year 2024-2025

1. What is the purpose of the get_grade() function?

The get_grade() function prompts the user to enter a grade, converts it to a floating-point number, and ensures it does not exceed 100. If the entered grade is greater than 100, the program displays "Invalid Grade" and exits using sys.exit().

2. How does the program handle the summary of grades?

After calculating the Prelim, Midterm, and Final grades, the program prints a summary of the student's grades, displaying their name along with the computed grades for each term. It also attempts to convert the final grade into a numerical grading system based on UCC numerical grading system.

7. Conclusion:

In conclusion, the goal of incorporating literals and variables into a Python program was successfully achieved. The program was able to accurately calculate the final grades for a CpE course by using basic math operations and conditional statements to ensure that the inputs were within the correct range. To test the program, data for three students was input, and it correctly displayed each student's Prelim, Midterm, and Final grades, which were then converted to the UCC grading system. This exercise provided a hands-on opportunity to learn how to use literals and variables effectively, while also applying basic mathematical operations and conditional logic in Python. It reinforced the understanding of how to write a program that processes and validates input to produce the desired output.

8. Assessment Rubric