



Lab 5 - Mathematical Functions and Strings

Chapter:	3. Mathematical Functions and Strings
Time:	80 Minutes

Lab
5

Objectives

- To familiarize students how to solve practical problems programmatically.
- To solve mathematics problems by using the functions in the math module.
- To represent and process strings and characters.
- To represent special characters using the escape sequence.
- To invoke the print function with the end argument.
- To convert numbers to a string using the str function.
- To use the + operator to concatenate strings.
- To read strings from the keyboard.

Current Lab Learning Outcomes (LLO)

By completion of the lab the students should be able to

- Convert a simple mathematical equation into a Python expression.
- Use the *eval* function to evaluate and convert a string to a numerical value.
- Use the *import* keyword to load a module into a program.
- Use the constants of the *math* module such as pi.
- Use the functions of *math* module such as the *sqrt* and *sin* functions.
- Use the *round* function.
- Use the special character (`\n`) for inputting a new line into a string.
- Use plus sign (+) to concatenate strings.

Lab Requirements

- PyCharm (IDE).

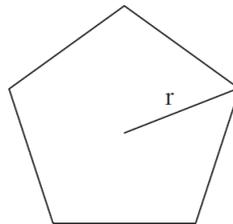


Practice Activities with Lab Instructor (20 minutes)

Problem 1

Programming Exercises (3.1)

Write a program that prompts the user to enter the length from the center of a pentagon to a vertex and computes the area of the pentagon, as shown in the following figure.



The formula for computing the area of a pentagon is $Area = \frac{3\sqrt{3}}{2}s^2$, where s is the length of a side. The side can be computed using the formula $s = 2r \sin \frac{\pi}{5}$, where r is the length from the center of a pentagon to a vertex.

```
Enter the length from the center to a vertex: 5.5 <enter>  
The area of the pentagon is 108.61
```

Solution

Phase 1: Problem-Solving Phase:

- 1- Ask the user to enter the length from the center to a vertex (r).
 - o `r = eval(input("message..."))`
- 2- Calculate the length of the side (s).
 - o Use the `sin` function in the math module to compute ($\sin \frac{\pi}{5}$)
 - o Use the `PI` constant in the math module to get the value of π
 - o `s = (2 * r) * math.sin(math.PI / 5)`
- 3- Calculate the area.
 - o Use the `sqrt` function in the math module to compute ($\sqrt{3}$)
 - o Use the `pow` function (`pow(s, 3)`) or (`s ** 3`) to compute (s^2)
 - o `area = ((3 * math.sqrt(3)) / 2) * pow(s, 2)`
- 4- Display the result (`area`).

Phase 2: Implementation Phase:

1. Create a new project and name it "Lab 5".

2. Create a new file and name it "activity_1.py".
3. Write the following code in the file:

activity_1.py

```
1 import math
2
3 # Get the length from the center to a vertex
4 r = eval(input("Enter the length from the center to a vertex: "))
5
6 # Calculate the length of the side
7 s = 2 * r * math.sin(math.pi / 5)
8
9 # Calculate the area
10 area = 3 * math.sqrt(3) * s * s / 2
11
12 # Display the result
13 print("The area of the pentagon is", round(area, 2))
```

Problem 2

Programming Exercises (3.1)

Write a program that reads the following information and prints a payroll statement:

- Employee's name (e.g., Smith)
- Number of hours worked in a week (e.g., 10)
- Hourly pay rate (e.g., 9.75)
- Federal tax withholding rate (e.g., 20%)
- State tax withholding rate (e.g., 9%)

A sample run is shown below:

```
Enter employee's name: Smith <enter>
Enter number of hours worked in a week: 10 <enter>
Enter hourly pay rate: 9.75 <enter>
Enter federal tax withholding rate: 0.20 <enter>
Enter state tax withholding rate: 0.09 <enter>
```

```
Employee Name: Smith
Hours Worked: 10.0
Pay Rate: $9.75
Gross Pay: $97.5
Deductions:
    Federal Withholding (20.0%): $19.5
    State Withholding (9.0%): $8.77
    Total Deduction: $28.27
Net Pay: $69.22
```

Solution

Phase 1: Problem-Solving Phase:

- 1- Ask the user to enter the employee's name (**name**).
 - o the name is a string value, so you shouldn't use the eval function.
 - o `name = input("message...")`
- 2- Ask the user to enter the number of hours worked in a week (**hours**).
 - o `hours = eval(input("message..."))`
- 3- Ask the user to enter the hourly pay rate (**payRate**).
 - o `payRate = eval(input("message..."))`
- 4- Ask the user to enter the federal tax withholding rate (**fedTaxWithholdingRate**).
 - o `fedTaxWithholdingRate = eval(input("message..."))`
- 5- Ask the user to enter the state tax withholding rate (**stateTaxWithholdingRate**).
 - o `stateTaxWithholdingRate = eval(input("message..."))`
- 6- Calculate the gross Pay (**grossPay**).
 - o `grossPay = hours * payRate`
- 7- Calculate the federal Withholding (**stateTaxWithholding**).
 - o `fedTaxWithholding = grossPay * fedTaxWithholdingRate`
- 8- Calculate the state Withholding (**stateTaxWithholding**).
 - o `stateTaxWithholding = grossPay * stateTaxWithholdingRate`
- 9- Calculate the total deduction (**totalDeduction**).
 - o `totalDeduction = fedTaxWithholding + stateTaxWithholding`
- 10- Calculate the net pay (**netPay**).
 - o `netPay = grossPay - totalDeduction`
- 11- Store, prepare and format the output into a string (**out**).
 - o You can concatenate strings using plus sign (+).
 - o You can write this (`out += "appended string"`) instead of this (`out = out + "appended string"`)
 - o You can convert a numerical value to a string by using the *str* function.
 - o You can use the "\n" special character to insert a new line to a string.
 - o You can use the *int* function to convert a float or string value to integer.
 - o You can use the "\ " continuation symbol to write a statement in multiple lines.
 - o `percentage = rate * 100`
- 12- Display the result (**out**).

Phase 2: Implementation Phase:

1. Open the project "Lab 5" if it was not opened or create it if it was not existing.
2. Create a new file and name it "activity_2.py".
3. Write the following code in the file:

activity_2.py

```
1  # Get the employee's name
2  name = input("Enter employee's name: ")
3
4  # Get the number of hours worked in a week
5  hours = eval(input("Enter number of hours worked in a week: "))
6  # Get the hourly pay rate
7  payRate = eval(input("Enter hourly pay rate: "))
8  # Get the federal tax withholding rate
9  fedTaxWithholdingRate = eval(input("Enter federal tax withholding rate: "))
10 # Get the state tax withholding rate
11 stateTaxWithholdingRate = eval(input("Enter state tax withholding rate: "))
12
13 # Calculate the gross Pay
14 grossPay = hours * payRate
15 # Calculate the federal Withholding
16 fedTaxWithholding = grossPay * fedTaxWithholdingRate
17 # Calculate the state Withholding
18 stateTaxWithholding = grossPay * stateTaxWithholdingRate
19 # Calculate the total deduction
20 totalDeduction = fedTaxWithholding + stateTaxWithholding
21 # Calculate the net pay
22 netPay = grossPay - totalDeduction
23
24 # Store, prepare and format the output into a string
25 out = "Employee Name: " + name + "\n\n"
26 out += "Hours Worked: " + str(hours) + '\n'
27 out += "Pay Rate: $" + str(payRate) + '\n'
28 out += "Gross Pay: $" + str(grossPay) + '\n'
29 out += "Deductions:\n"
30 out += "  Federal Withholding (" + str(fedTaxWithholdingRate * 100) + \
31     "%): $" + str(int(fedTaxWithholding * 100) / 100.0) + '\n'
32 out += "  State Withholding (" + str(stateTaxWithholdingRate * 100) + "%):" + \
33     " $" + str(int(stateTaxWithholding * 100) / 100.0) + '\n'
34 out += "  Total Deduction:" + " $" + \
35     str(int(totalDeduction * 100) / 100.0) + '\n'
36 out += "Net Pay:" + "   $" + str(int(netPay * 100) / 100.0)
37
38 # Display the result
39 print(out)
```



Individual Activities (60 minutes)

Problem 3

Programming Exercises (3.4)

The area of a pentagon can be computed using the following formula (s is the length of a side):

$$Area = \frac{5 \times s^2}{4 \times \tan\left(\frac{\pi}{5}\right)}$$

Write a program that prompts the user to enter the side of a pentagon and displays the area. Here is a sample run:



```
Enter the side: 5.5 <enter>  
The area of the pentagon is 52.04444136781625
```

Problem 4

Programming Exercises (3.11)

Write a program that prompts the user to enter a four-digit integer and displays the number in reverse order. Here is a sample run:



```
Enter an integer: 3125 <enter>  
The reversed number is 5213
```

Extra Exercises (**Homework**)

From the Textbook

- Programming Exercises:
 - 3.2
 - 3.3
 - 3.5
 - 3.8

From MyProgrammingLab (<https://pearson.turingscraft.com>)

- 3.3
 - 51835
 - 51755
 - 51756
 - 51839
 - 51869
- 3.6
 - 51028

Upload Your Solutions



Upload your solutions of the lab activities to Blackboard.