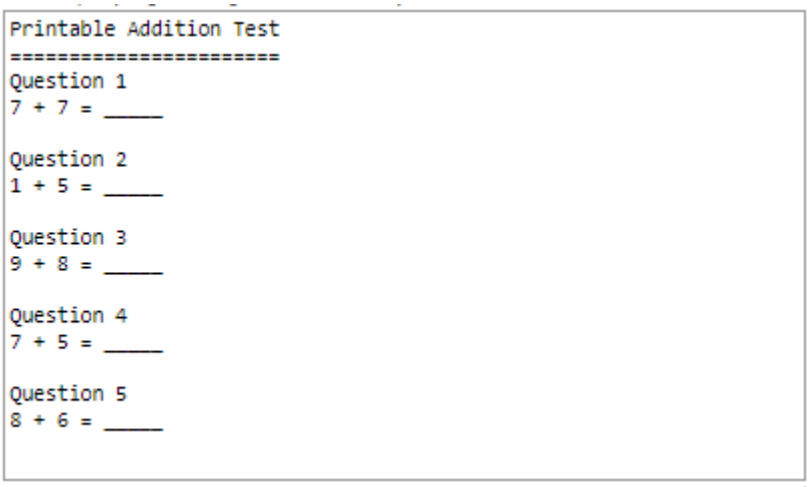
PMC2306 - Programming lab in Python

LAB CYCLE 2

(Programs based on **Function,List and String**)

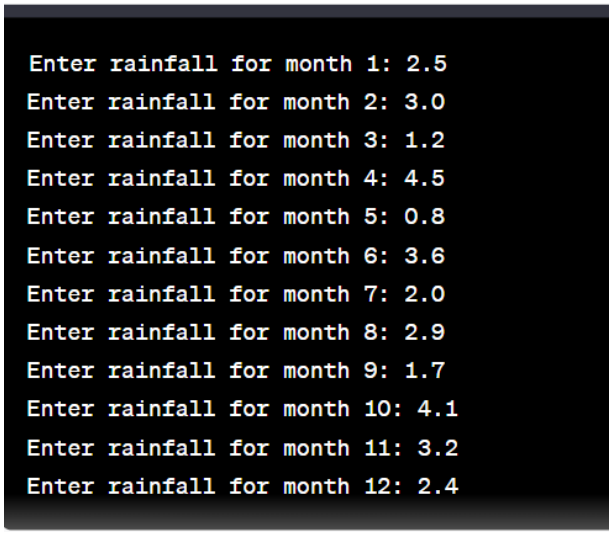
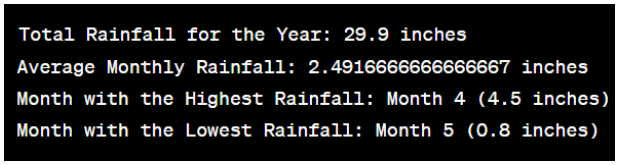
<table>
<tr><td>11</td><td>Write a program that generates printable addition tests. The tests should consist of 5 questions which present a simple addition question in the following format, where the question number goes from 1 to 5, and num1 and num2 are randomly generated numbers between 1 and 10:
Question 1
num1 + num2 = _____
The program should simply display the 5 questions; it should not prompt the user for any input.
SAMPLE OUTPUT
 
</td></tr>
<tr><td>12</td><td>Write a program that asks the user to enter five test scores. The program should display a letter grade for each score and the average test score. Write the following functions in the program:
• calc_average. This function should accept five test scores as arguments and return the average of the scores.
• determine_grade. This function should accept a test score as an argument and return a letter grade for the score based on the following grading scale:
<table>
<tr><th>Score</th><th>Letter Grade</th></tr>
<tr><td>90–100</td><td>A</td></tr>
<tr><td>80–89</td><td>B</td></tr>
<tr><td>70–79</td><td>C</td></tr>
<tr><td>60–69</td><td>D</td></tr>
<tr><td>Below 60</td><td>F</td></tr>
</table></td></tr>
</table>

	SAMPLE OUTPUT Enter the grade for test 1: 56 Enter the grade for test 2: 96 Enter the grade for test 3: 75 Enter the grade for test 4: 88 Enter the grade for test 5: 65 Score Letter Grade ------------------------- 56 F 96 A 75 C 88 B 65 D Your average is 76.00
13	Write a Boolean function named **is_prime** which takes an integer as an argument and returns true if the argument is a prime number, or false otherwise. Use the function in a program that prompts the user to enter a number then displays a message indicating whether the number is prime. **SAMPLE OUTPUT** Please enter a number to check if it is prime or not: 13 Your number 13 is a prime number.
14	Suppose you have taken out a loan for a certain amount of money with a fixed monthly interest rate and monthly payments, and you want to determine the monthly payment amount necessary to pay off the loan within a specific number of months. The formula is as follows: $$P = \frac{R * A}{1 - (1 + R)^{-M}}$$ The terms in the formula are: • P is the payment amount per month. • R is the monthly interest rate, as a decimal (e.g., 2.5% 5 0.025). • A is the amount of the loan. • M is the number of months. Write a program that prompts the user to enter the monthly interest rate as a percentage, the loan amount, and the desired number of months. The program should pass these values to a function that returns the monthly payment amount necessary. The program should display the loan amount, interest rate, payment months, and monthly payment amount.

	Sample Output:
	`Loan Payment Calculator`
	`======================`
	`Enter the loan amount: $10000`
	`Enter the annual interest rate (%): 12`
	`Enter the number of months: 24`
	
	`Loan Details:`
	`Principal Loan Amount: $10000.00`
	`Annual Interest Rate: 12.0%`
	`Number of Months: 24`
	`Monthly Payment Amount: $470.73`

15	Write a program that lets the user play the game of Rock, Paper, Scissors against the computer.
	The program should work as follows:
	1. When the program begins, a random number in the range of 1 through 3 is generated. If the number is 1, then the computer has chosen rock. If the number is 2, then the computer has chosen paper. If the number is 3, then the computer has chosen scissors. (Don't display the computer's choice yet.)
	2. The user enters his or her choice of "rock," "paper," or "scissors" at the keyboard.
	3. The computer's choice is displayed.
	4. A winner is selected according to the following rules:
	• If one player chooses rock and the other player chooses scissors, then rock wins. (Rock smashes scissors.)
	• If one player chooses scissors and the other player chooses paper, then scissors wins. (Scissors cuts paper.)
	• If one player chooses paper and the other player chooses rock, then paper wins. (Paper wraps rock.)
	• If both players make the same choice, the game must be played again to determine the winner.
	
	SAMPLE OUTPUT
	`Let's play Rock, Paper, Scissors!`
	`Enter your choice (rock, paper, scissors): rock`
	`Computer chooses: paper`
	`Computer wins!`
	`Do you want to play again? (yes/no): yes`
	`Enter your choice (rock, paper, scissors): paper`
	`Computer chooses: paper`
	`It's a tie!`
	`It's a tie! Play again.`

16	Design a program that generates a seven-digit lottery number. The program should generate seven random numbers, each in the range of 0 through 9, and assign each number to a list element. (Random numbers were discussed in Chapter 5.) Then write another loop that displays the contents of the list.

	Sample Output The lucky numbers are below. Thanks for participating. 6 6 0 4 8 7 6
17	Design a program that lets the user enter the total rainfall for each of 12 months into a list. The program should calculate and display the total rainfall for the year, the average monthly rainfall, the months with the highest and lowest amounts. Sample Output  

18	Design a program that asks the user to enter a series of 6 numbers. The program should store the numbers in a list then display the following data:
	• The lowest number in the list
	• The highest number in the list
	• The total of the numbers in the list
	• The average of the numbers in the list
	`Enter number 1: 45`
	`Enter number 2: 12`
	`Enter number 3: 67`
	`Enter number 4: 89`
	`Enter number 5: 100`
	`Enter number 6: 34`
	
	`Lowest number: 12.0`
	`Highest number: 100.0`
	`Total of the numbers: 347.0`
	`Average of the numbers: 57.833333333333336`

19	At the university, passwords for the campus computer system must meet the following requirements:
	• The password must be at least seven characters long.
	• It must contain at least one uppercase letter.
	• It must contain at least one lowercase letter.
	• It must contain at least one numeric digit.
	
	When a student sets up his or her password, the password must be validated to ensure it meets these requirements. You have been asked to write the code that performs this validation. You decide to write a function named valid_password that accepts the password as an argument and returns either true or false, to indicate whether it is valid.
	`Enter your password: bozo Enter`
	`That password is not valid.`
	`Enter your password: kangaroo Enter`
	`That password is not valid.`
	`Enter your password: Tiger9 Enter`
	`That password is not valid.`
	`Enter your password: Leopard6 Enter`
	`That is a valid password.`

20	The process of breaking a string into tokens is known as tokenizing a string. In Python, you use the split method to tokenize strings.
	# Strings to tokenize
	str1 = 'one two three four'
	str2 = '10:20:30:40:50'
	str3 = 'a/b/c/d/e/f'
	
	Sample Output
	Token: one
	Token: two
	Token: three
	Token: four
	
	Token: 10
	Token: 20
	Token: 30
	Token: 40
	Token: 50
	
	Token: a
	Token: b
	Token: c
	Token: d
	Token: e
	Token: f