

Marking Scheme for COMP Prelim Practical Exam P2 2020

Name : _____ Class: _____

Task1 (10)	Task2 (10)	Task3 (10)	Task4 (20)	Total (50)

Task 1 – Simple program / String Manipulation

No	Description
1.	= ROUND(MEDIAN(D4:D18),0)
2.	= LEFT(A4,1) to LEFT(A18,1)
3.	= VLOOKUP(E4,\$C\$25:\$D\$28,2,TRUE)
4.	=FV(G4,E4,0,-D4)
5.	=IF(F4="S",0.1*H4,0)
6.	=H4-I4
7.	=COUNTIF(F4:F18,"S")

Task 2 – Modify/Edit existing program

8.	username = name + (10-size)*'9'
9.	a) name = input("Please enter your name: ") size = len(name) print("Name : ", name) while (size < 3 or size > 20) or not name.isalpha(): print("Error : Name must be between 3 to 20 and contain characters only") name = input("Please enter your name: ") size = len(name)
	b) digit = int(input("Please enter a single digit (0 to 9): ")) while digit not in [0,1,2,3,4,5,6,7,8,9]: print("Error : Single digit (0 to 9) only") digit = input("Please enter a single digit (0 to 9): ") if size < 10: username = name + (10-size)*str(digit) else: username = name[0:10]

Task 3 – Identify and correct errors in existing program

10.	#list of fruits sorted in ascending order
-----	---

	<pre> fruits_list = ["apple", "grape", "guava", "mango", "orange", "pear"] fruit_to_find = input("Which fruit would you like to search for? ") # items = length(fruits_list) # found = True # for i in range(item): # if fruits_list[index] == fruit_to_find: # print("There are " + str(items) + " fruits in the list, " + fruit_to_find + " is item " + str(i) + " in the list.") # found = True else fruits_list[i] < fruit_to_find: # # break if found = True: # # print("The fruit is not in the list.") </pre>
Task 4 – Develop/Write a program	
11.	<pre> numstr = input("Enter a 12-digit number : ") size = len(numstr) while size != 12 or not numstr.isdigit(): if size != 12: print("Error : Must be exactly 12 digits") if not numstr.isdigit(): print("Error : Only digits are allowed") numstr = input("Enter a 12-digit number : ") size = len(numstr) even_sum=0 odd_sum=0 for index in range(size): digit = int(numstr[index]) if (index + 1) % 2 == 0: even_sum += digit else: odd_sum += digit check_digit = (odd_sum * 3 + even_sum) % 10 if check_digit != 0: check_digit = 10 - check_digit print("{} is the check digit for the 12-digit number {}").format(check_digit,numstr)) </pre>
12.	Test : bitmap output matches stored program
	five lines output (-1 for each line with an error)
13.	Extend : correct test for both odd and even weights
	correct output (-1 for each line with an error)
14.	Extend : check for 12-digit or 13-digit
	correct output (-1 for each line with an error)
	Total