



Programming I

Year 1 (2023/24), Semester 1

SCHOOL OF INFOCOMM TECHNOLOGY

Diploma in Information Technology
Diploma in Data Science
Diploma in Cybersecurity & Digital Forensics
Diploma in Immersive Media
Common ICT Programme

COMMON TEST

Date: 5 Jun 2023 (Monday)

Time: 11.00 AM – 12.30 PM

INSTRUCTIONS TO CANDIDATES:

1. This paper consists of 7 pages including this cover page. Check carefully to make sure your set is complete.
2. There are 5 questions. Answer ALL questions.
3. You are allowed to use calculators and one computer.
4. You may refer to soft copy/hard copy of PRG1 teaching materials and programs. **However, you are not allowed to access the Internet.**

Instructions

Prior to test

- Create a folder `CT_S12345678` on your desktop (replace `S12345678` with your student id)
- Create 5 Python files in that folder with the following naming convention
 - a. `CT_Q1_S12345678.py`
 - b. `CT_Q2_S12345678.py`
 - c. `CT_Q3_S12345678.py`
 - d. `CT_Q4_S12345678.py`
 - e. `CT_Q5_S12345678.py`

replace `S12345678` with your student id
- In each file, enter your **name**, **id** and **group** in the first line as comment:
e.g. `# John Tan (S12345678) - IT01`

Submission

- In **POLITEMail**, navigate to **PRG1 > Assessment > Common Test Submission**
 - Compress the folder containing the 5 Python files into a **.zip** file
 - Upload **.zip** file and click the "**Submit**" button

Note: It is **your RESPONSIBILITY** to ensure that the files are submitted correctly

PLAGIARISM WARNING:

If a student is found to have submitted work not done by him/her, he/she will not be awarded any marks for this practical test. Disciplinary action may also be taken.

Similar action will be taken for student who allows other student(s) to copy his/her work, or posting any solutions or code related to the practical test before the end of the hour for the test.

QUESTION 1 (20 marks)

A fried rice stall allows customers to customise their preferred dish with different toppings, and to generate a quote for the amount payable. It is optional for customers to purchase their dish with toppings.

The base 'Jasmine egg fried rice' is \$2.50.

It costs \$1.20 for one portion of toppings which contains some fixed items.

Customer is entitled to 10% discount off the total bill for having the stall's loyalty card.

The stall owner has written a Python program as shown in Figure 1(a).

```
require_var = int(input('Do you require toppings? [Yes / No]: '))
toppings_set_var = input('Portions of toppings required: ')
loyalty_var = input('Do you have the \'loyalty\' card? [Yes / No]: ')

total_cost = 2,5

if require_var.lower() == "Yes":
    total_cost = toppings_set_var * (1.2 + total_cost)

if loyalty_var.capitalize() == "Yes"
    total_cost x= 0.9

print("Total cost: ${:.2f}".format(total_cost))
```

Figure 1(a): Python program

There are some errors in the program. Help the stall owner debug and resolve the errors.

Copy the program code from Figure 1(a) into your Python file that you have created for this question and edit the program so that it is able to achieve the output shown in Figure 1(b).

```
Do you require toppings? [Yes / No]: Yes
Portions of toppings required: 4
Do you have the 'loyalty' card? [Yes / No]: Yes
Total cost: $6.57
```

Figure 1(b): Sample output (values underlined are user input)

QUESTION 2 (20 marks)

Tom would like to create a word scramble game that swaps all occurrences of 2 characters at the selected positions.

Write a Python program to perform the following:

- Prompt the user for a word in lowercase
- Display the number of characters in the word
- Ask user to input 2 numbers that selects 2 characters within the word (you may assume that user input is always within the range)
- Print out the two characters selected
- Swap all occurrences of the selected characters
- Print the scrambled word

Sample Runs (the input values are underlined):

```
Enter your word: programming
Your input has 11 characters
Input the positions of the characters to select, separated by ',': 3,4
Swapping the characters 'o' and 'g'
Scrambled word: prgorammno
```

Figure 2(a): Sample output for first run (values underlined are user input)

```
Enter your word: positivity
Your input has 10 characters
Input the positions of the characters to select, separated by ',': 4,5
Swapping the characters 'i' and 't'
Scrambled word: postitvtiy
```

Figure 2(b): Sample output for second run (values underlined are user input)

QUESTION 3 (20 marks)

The City Library uses the following table to calculate the fine when books are returned late:

Fine calculation		Number of Books (y)	
		1	more than 1
Number of days late (x)	1	1.20	$0.15y + 1.20$
	more than 1	$(2^{x-1}) 0.3 + 1$	$(2^x) 0.15 y + 1.20$

Write a Python program to perform the following:

- Prompt the user for the values of x and y (you may assume that the input values are positive integer numbers)
- Calculate the fine based on the values of x and y entered
- Display an appropriate message showing the fine

Sample outputs are shown below:

```
Please enter the number of books that are late: 1
Please enter the number of days the books are late: 1
The fine for 1 book(s) for 1 day(s) is $1.20
```

Figure 3(a): Sample output for first run (values underlined are user input)

```
Please enter the number of books that are late: 4
Please enter the number of days the books are late: 1
The fine for 4 book(s) for 1 day(s) is $1.80
```

Figure 3(b): Sample output for second run (values underlined are user input)

```
Please enter the number of books that are late: 1
Please enter the number of days the books are late: 4
The fine for 1 book(s) for 4 day(s) is $3.40
```

Figure 3(c): Sample output for third run (values underlined are user input)

```
Please enter the number of books that are late: 4
Please enter the number of days the books are late: 4
The fine for 4 book(s) for 4 day(s) is $10.80
```

Figure 3(d): Sample output for fourth run (values underlined are user input)

QUESTION 4 (20 marks)

Tom made the following purchases at a bakery shop:

Item Name	Unit Price (\$)	Quantity
Curry puff	2.40	2
Apple tart	2.00	4
Tuna puff	2.20	5
Egg tart	1.80	1
Custard tart	1.50	2

Write a Python program to do the following:

- Create a **nested** list to store the data given in the table, with each element as a list to store the 3 values (Item Name, Unit Price, and Quantity) for an item.
- Calculate the cost of puffs and the cost of tarts separately.
- Display the details of the purchases as shown in Figure 4.

Item Name	Unit Price	Quantity	Amount
Curry puff	2.40	2	4.80
Apple tart	2.00	4	8.00
Tuna puff	2.20	5	11.00
Egg tart	1.80	1	1.80
Custard tart	1.50	2	3.00
Total cost of tarts: \$12.80			
Total cost of puffs: \$15.80			

Figure 4: Sample output

You are required to use a **loop** in your program.

QUESTION 5 (20 marks)

Tom wants to implement a counting game program.

Write a Python program to do the following:

- Prompt user for two integers num1 and num2
- Prompt user for a positive integer, x, between num1 and num2 inclusive
- Display the numbers from num1 to num2 in ascending order, replacing all integers which are divisible by x with the word “skip”
- Display the number of integers skipped

For example, if x is 7, numbers divisible by 7 such as 7, 14, 21, ... will not appear but will be printed as “skip”.

The program should ensure that num1 is smaller than num2. If num1 is bigger than num2, the values of num1 and num2 should be swapped. It should also check that the integer x is between num1 and num2 inclusive and proceed only if the input is valid.

You are required to use **loop** and **selection** statements in the program and follow the formatting in the sample output shown in Figure 5.

```
Enter num1: 1
Enter num2: 100
Enter a number between 1 and 100: 101
Invalid number entered
```

Figure 5(a): Sample output for first run (values underlined are user input)

```
Enter num1: 3
Enter num2: 50
Enter a number between 3 and 50: 7
3
4
5
6
skip
8
9
10
11
12
13
skip
15
.
.
.
48
skip
50
7 integers skipped
```

Figure 5(b): Sample output for second run (values underlined are user input)