

# Jurong West Secondary School

# **Preliminary Examinations 2018**

# COMPUTING

Secondary Four Express Paper 2 Practical (Lab-based) Additional Materials: Electronic version of INS.XLSX file Electronic version of WORDS.PY file Electronic version of AVERAGE.PY file Insert Quick Reference Glossary

28 August 2018 0800 – 1030 2 hours 30 minutes

7155/02

## **READ THESE INSTRUCTIONS FIRST**

Write your name, class and index number in the spaces provided at the top of this page. Write in dark blue or black pen.

Do not use staples, paper clips, glue or correction fluid.

Answer all questions.

All tasks must be done in the computer laboratory. You are not allowed to bring in or take out any pieces of work or materials on paper or electronic media or any other form.

Programs are to be written in Python.

Save your work using the file name given in the question as and when necessary.

The number of marks is given in brackets [] at the end of each question or part question. The total number of marks for this paper is 50.

After checking of answer script						
Checked by	Signature	Date				
Student						

For Examiner's Use				
Task 1				
Task 2				
Task 3				
Task 4				
Total				

This document consists of <u>7</u> printed pages.



# Answer all questions

# Task 1 (10 Marks)

Great Western Assurance uses a spreadsheet for its employee sales records. You are required to finish setting up the spreadsheet.

#### Open the file **INS.xlsx**

You will see the following data.

$\mathcal{A}$	A	В	С	D	E	F	G	Н	
1	GREAT WESTERN ASSURANCE								
2	Employee Sales Records								
3									
4									
5	Insurance sales in Augu								
6									
7	Transaction No.	Name of Sales Agent	Employee ID	Gender	Type of Insurance	Bonus Awarded	No. of Policies Sold	Total Bonus	
8		AYSHAM RASHID	2018M2309353		LIFE		3		
9		BALASUNDARI MAMANNAN	2018M3253465		MEDICAL		10		
10		CHONG KE ER	2018F2534676		MEDICAL		5		
11		ELYSE ONG YU XUAN	2018F2351214		LIFE		5		
12		GAN SIEW SHI	2018F6474342		VEHICLE		4		
13		KOH WEN QI	2018F3456457		FIRE		2		
14		LEONG JUN TING	2018F4567562		VEHICLE		4		
15		MOHAMMAD SUFIAN	2018M3466472		MEDICAL		9		
16		REUBEN YAP JING XIANG	2018M1986543		LIFE		12		
17		THARUN S/O THAYALAN	2018F2534656		LIFE		2		
18									
19	Commission Table						Summary Information (Partial)		
20		Policies Sold	1	5	11		Range of No. of Policies Sold		
21		Description	One to Four	Five to Ten	Eleven and above		Number of Female Agents		
22		Percentage Commission Rate	100%	105%	110%		Total bonus of all agents		
23							Was the target met?		
24	Bonus Details Table								
25		Type of Insurance	Insurance Name	Bonus					
26		FIRE	Fire insurance	\$3,500					
27		LIFE	Life insurance	\$5,000					
28		MEDICAL	Medical insurance	\$1,500					
29		VEHICLE	Vehicle insurance	\$4,000					

Save the file as **MYINS\_**<class>\_<index number>\_<your name>.

1	In the <b>Transaction No.</b> column, enter a function to generate a random number between 1 and 888 inclusive.	[1]
2	For the cell range <b>D8:D17</b> , extract the fifth letter of the <b>Employee ID</b> to represent the <b>Gender</b> .	[1]
3	In cell <b>H20</b> , enter a formula to calculate the range of the <b>No. of Policies</b> Sold.	[1]
4	In cell <b>H21</b> , enter a formula to calculate the number of female agents.	[1]
5	Use an appropriate function to search for the <b>Bonus</b> in the <b>Bonus Details</b> <b>Table</b> and use it to complete the <b>Bonus Awarded</b> column.	[1]
6	To show <b>Total Bonus</b> in the cell range <b>H8:H17</b> ,	
	<ul> <li>use an appropriate function to search for the Percentage Commission Rate in the Commission Table,</li> </ul>	

• and multiply it by both the Bonus Awarded and No. of Policies Sold. [2]

- 7 In cell **H22**, enter a formula to calculate the total bonus of all the agents. [1]
- 8 In order to check whether the target was met for the month of August, two conditions need to be met:
  - the total bonus of all the agents must exceed \$190,000;
  - the range of the no. of policies sold must be less than 8.

Using a conditional statement, identify whether the target was met, and put **YES** in the cell **H23**. Otherwise, put **NO** in cell **H23**. [2]

Save and close your file.

# Task 2 (10 marks)

The following program accepts 7 alphanumeric words from the user and separates the digits and alphabets. The separated digits are then totalled before printing.

```
num_words = 7
for i in range(num_words):
    word = input("Enter an alphanumeric word: ")
    num_total = 0
    for ch in word:
        if ch.isnumeric():
            num_total = num_total + int(ch)
    print("The total value of the digits in the word is", num_total)
```

Open the file **WORDS.py** 

Save the file as MYWORDS\_<class>\_<index number>\_<your name>.

- **9** Edit the program so that it:
  - (a) Allows only 3 words to be entered. [1]
  - (b) Prints out the concatenated string of alphabets as well. [4]

```
Sample Output:
Enter an alphanumeric word: al9hg6!
The total value of the digits in the word is 15
The concatenated alphabets in the word is alhg
```

(c) Checks if the word input provided by the user is made up of more than 5 characters and if not, asks the user to input again, as necessary, informing them about the requirements.

Save your program.

10 Save your program as VARWORDS\_<class>\_<index number>\_<your name>.

Edit your program so that it does not limit the number of words that are entered.

Save and close your file.

[3]

[2]

# Task 3 (10 Marks)

Knowing your expertise in the field of mathematics, you have been asked to write a program that will input several integers into a list and find the average of the integer elements in the list. If there are no elements in the list, the average will be 0.

There are several syntax errors and logical errors in the program.

```
values = []
while True
    input str = input("Enter integer, blank to end: )
    if input str = "":
        break
    values.prepend(int(input_str))
if len(Values) > 0:
   total = 0
     index = 0
    while index < len[values]:</pre>
        total = total + values[index]
        index += 1
       average = total * len(values)
elif:
    average = 0
print("The average of the given inputs is { ".format(average))
```

Open the file **AVERAGE.py** Save the file as **MYAVERAGE** \_<Class>\_<Index>\_<Name>.

**11** Identify and correct the errors in the program so that it works correctly.

[10]

Save and close your file.

# Task 4 (20 Marks)

In the year 2029, global warming has begun to affect consumer items, namely eggs. A company wishes to employ data-mining to extract information such as the average number of eggs spoilt and which of their packing layers need maintenance. Presently, each clutch of eggs is packed in five layers. You have been employed to write a suitable program to assist the company in their objective.

The program should allow you to:

- Enter data in the format a:b:c:d:e where a, b, c, d and e are the number of spoilt eggs in each of the **five layers** of packaging.
  - An example would be 2:1:4:1:3, where 2 is the number of eggs spoilt in Layer 1, 1 in Layer 2 and so on.
- Only allow data entry values greater than or equal to 0.
- Calculate the total number of spoilt eggs in all the layers.
  - For the example 2:1:4:1:3, the total would be 11.
- Repeat this for a total of six shipments, assuming the same five layers are utilised during each shipment.
- Calculate the total number of spoilt eggs for the six shipments.
- Find the average number of spoilt eggs per layer, across the six shipments, rounded to the nearest whole number.
- Display this on the screen. Your output **must** look like this:

```
Shipment Report:

Shipment 1 11 eggs spoilt

Shipment 2 13 eggs spoilt

Shipment 3 20 eggs spoilt

Shipment 4 14 eggs spoilt

Shipment 5 17 eggs spoilt

Shipment 6 18 eggs spoilt

Total number of spoilt eggs 93

Average number of spoilt eggs per layer 19
```

**12** Write your program and test that it works.

[10]

Save your program as EGG1\_<class>\_<index number>\_<your name>.

6

- **13** When your program is working, use the following test data to show your test results.
  - 2:1:4:1:3 1:2:3:4:3 4:2:4:5:5 3:3:3:2:3 2:4:4:3:4 3:3:2:4:6

Take a screen shot of your results and save it as a bitmap **EGGRESULTS**\_<class>\_<index number>\_<your name>.

14 Save your program as EGG2\_<class>\_<index number>\_<your name>. Extend your program to identify layers that require maintenance as they contain more than 19 spoilt eggs in total, for the six shipments. Print out the information as follows:

Layer 3 20 eggs spoilt Layer 5 24 eggs spoilt

Save your program.

**15** Save your program as EGG3\_<class>\_<index number>\_<your name>. Extend your program to work for any number of layers. [2]

Save and close your file.

# End of Paper

[5]

[3]