		Thumb drive No:	
Class/ Index Number	Centre Number/ 'O' Level Index Number	Name	
/	/		



新加坡海星中学 MARIS STELLA HIGH SCHOOL PRELIMINARY EXAMINATION SECONDARY FOUR

COMPUTING

Paper 2 Practical (Lab-based)

7155/02 20 August 2021 2 hours 30 mins

Additional Materials: Electronic version of REGISTER.XLSX file Electronic version of PRODUCT.PY file Electronic version of GUESS.PY file Insert Quick Reference for Python

READ THESE INSTRUCTIONS FIRST

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 in 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.



This document consists of 8 printed pages.

Your company is in charge of organising the National Coding Competition held every year. Students register using their names, birth years (year they were born) and their school codes. As the organiser, you have to create a unique password for each student so that each student is able to access his personal account space in the National Coding Competition website.

You are required to finish setting up the spreadsheet to create the unique passwords for the registered students.

	А	В	С	D	E	F	G	н	1	J	К	L	м	N
1	1 National Coding Competition Registration													
2	S.No	Name	Birth Year	Age	School Code	Category	Password				Category By Age			
3	1	Alden Zheng	2005		MSHS					Age Range	Age	Category		
4	2	Bryan See	2005		MVSS					12 to 13	12	Junior		
5	3	Chong Li	2004		BDSS					14 to 16	14	Senior		
6	4	Darius Khoo	2008		MSHS					17 and above	17	Open		
7	5	Dylan Ho	2008		MVSS									
8	6	Ethan Sim	2008		BDSS					Registration Nu	mber by Category		Fees Chargeable to	
9	7	Eugene Thong	2007		MSHS					Category	Student Total		MSHS	
10	8	Gabriel Ho	2009		MVSS					Junior			MVSS	
11	9	Ian Chua	2004		BDSS					Senior			BDSS	
12	10	lan Loh	2008		MSHS					Open				
13	11	Joshua Lai	2007		MVSS									
14	12	Jun Chian	2006		BDSS						Registr	ation Fees Chargal	ole	
15	13	Kenneth Teoh	2005		MSHS					School Code	Category	Cost per Category	Number Registered	Total
16	14	Marcus Ng	2006		MVSS					MSHS	Junior	\$ 3.00	0	
17	15	Nigel Neo	2009		BDSS					MSHS	Senior	\$ 5.00	0	
18	16	Owen Lok	2007		MSHS					MSHS	Open	\$ 8.00	0	
19	17	Rylan Ow	2008		MVSS					MVSS	Junior	\$ 3.00	0	
20	18	Kai Zee	2009		BDSS					MVSS	Senior	\$ 5.00	0	
21	19	Wei Quan	2004		MSHS					MVSS	Open	\$ 8.00	0	
22	20	Yan Kai	2006		MVSS					BDSS	Junior	\$ 3.00	0	
23	21	Zhen Tian	2009		BDSS					BDSS	Senior	\$ 5.00	0	
24										BDSS	Open	\$ 8.00	0	
25		Current Year	2021											

Open the file **REGISTER.xlsx** and you will see the following data.

Save the file as PASS_<class>_<index number>_<your name>.xlsx

- 1 In cells **D3** to **D23**, enter a formula that uses the **Current Year** and **Birth Year** to [1] calculate the **Age** of the student.
- 2 In cells **F3** to **F23**, enter a formula that uses an appropriate function to search for [2] the **Category** in the **Category By Age** table.
- 3 The unique password for each student is obtained using the following: [2] First 2 letters of the **Name** + length of **Name** + First letter of **Category**

e.g. If the record reads as follows Jackie Chan 2004 17 MVSS Open His password will be Ja11O

In cells **G3** to **G23**, use the function CONCATENATE with appropriate parameters to join different strings together to form the password.

- 4 In cells **K10** to **K12**, enter an appropriate function to calculate the total number [2] of students registered for each **Category**.
- 5 In cells **N16** to **N24**, enter a formula to calculate the total cost of [1] registration for each category for each school by multiplying the **Number Registered** by the **Cost per Category**.

6 In cells **N9** to **N11**, enter an appropriate function to calculate the total [2] cost chargeable to each school.

Save and close your file.

Task 2

The following program is used to determine if the product codes for a particular shop are valid. All product codes are 9 digits long. The first 8 digits are multiplied by a fixed weight based on their positions, then the total sum is calculated from the result of the product. This total sum is then divided by 10 and its remainder is known as the check digit. This check digit is compared with the last digit of the product code. If the two values are the same, the product code is considered as valid.

```
WEIGHTS=[3,4,6,8,4,3,5,5]
number = input("Please enter product number: ")
totalcount=0
for counter in range (8):
   totalcount = totalcount + WEIGHTS[counter]*int(number[counter])
mod = totalcount%10
if mod == int(number[-1]):
   print("The product number is valid.")
else:
   print("This is an invalid product number.")
```

Open the file PRODUCT.py

Save the file as MYPRODUCTCODE_<class>_<index number>_<your name>.py

- 7 Edit the program so that it:
 - (a) Divides the total sum by 11 to find the check digit. [1]
 - (b) Changes the check digit to the letter X if the remainder is 10 after dividing [2] by 11.
 - Accepts only a product code of length 9, where the first 8 characters are digits and the last character is either a digit or the letter X. A suitable error message must be displayed if an incorrect input is given. The program must loop until a valid product code is entered. This data input is case sensitive.

Save your program.

Edit your program so that it checks with the user if there is any other product [3] code to check. If the user enters a "Y" the program will start again until the user enters "N". A suitable message must be displayed to check for more product codes from the user. This data input should not case sensitive.

Save your program.

Task 3

The following program is a number guessing game that allows the user to guess a randomly generated number between 1 and 100 inclusive.

The program does the following:

- Generates a random number between 1 and 100 inclusive.
- Allows the user to repeatedly guess the randomly generated number by entering a number.
- Performs data validation to ensure the number entered is an integer.
- Allows the user to guess up to a maximum of 8 times.
- Informs the user that his guess is correct, if it matches the randomly generated number, and outputs an appropriate message.
- Otherwise, the program will inform the user that the correct number is either more than or less than his guess.
- When the number of tries reaches 8 and user fails to guess correctly, the program will output an appropriate end of game message.
- The program will end with the total number of tries the user used.

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

```
import random
count = 0
ans = random.randint(1,10)
num = input("Enter a number: ")
while not num.isdigit()
   num = input("Enter only numbers: ")
num=str(num)
while num != ans or count < 8:
    if num > ans:
       print("Your guess is too low!")
    else:
        print("Your guess is too high!")
    num = input("Enter a number: ")
    while not num.isdigit:
       num = input("Enter only numbers: ")
    num=str(num)
    count = count - 1
if num != ans:
    print("You guessed it correctly!")
else:
    print("You lost! You didn't get the number.")
```

```
print("You made {} guesses".format(num))
```

9 Open the file **GUESS.py**

[10]

Save the file as GUESSGAME_<class>_<index number>_<your name>.py

Task 4

It is the time of the year that the school needs to decide on Marist of the Year. Every class is required to elect one representative. All students are allowed to vote.

You have been asked to create a vote counting program.

Your program should include appropriate prompts for entry of data.

The program must:

- Allow the class form teacher to enter the name of the class.
- Allow the class form teacher to enter the number of students in the class.
- Allow the class form teacher to enter the number of candidates in the election up to a maximum of four candidates.
- Allow the class form teacher to enter the names of the candidates and store them in a suitable data structure.
- Allow each student to vote or abstain by entering a number. Ensure that this entry is validated.
- Count the total votes for each candidate and total student abstentions.

When all students have voted, display the name of the class, the votes for each candidate and the name of the candidate who has won the election. If there is a tie for first place, display all candidates with the equal highest number of votes.

An example of how the program should look like is as follows:

Name of class: 4C Number of students in class: 10 Number of candidates: 3 Enter names of candidates separated with a comma and a space: Ian, James, Keith

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 1

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 2

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 3

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 4

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 1 Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 2

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 3

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 1

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 2

Each student will now place their votes from the following choice: ['lan', 'James', 'Keith', 'Abstain'] Enter your choice from 1 to 4: 4

4C lan 3 votes James 3 votes Keith 2 votes ['lan', 'James'] won the election.

10 Write your program and test that it works. [13]

Save your program as CLASSVOTE1_<class>_<index number>_<your name>.py

- 11 When your program is working, test it for the following: [3]
 - Class 4C
 - Number of students in class 16
 - No of candidates 3
 - Names of candidates James, Jeffery, Jordan

Use the following numbers as votes

1	1	2	2
1	1	1	2
3	4	2	1
4	3	2	2

Take a screenshot of:

- The entry of starting details which includes two voters' data input. Save this screenshot as **TEST1_<class>_<index number>_<your name>**
- The screenshot of the result after all 16 votes have been entered. Save this screenshot as TEST2_<class>_<index number>_<your name>

Save your files either in .png or .jpg format.

- 12 Save your program as CLASSVOTE2_<class>_<index number>_<your [4] name>.py. Extend your program to:
 - In the event of a tie, allow the election to be run again with the tied candidates as the only remaining candidates. Make sure there is an appropriate message to inform the user there will be re-voting. All the students from the class should vote again, until there is only one winner with the highest votes. Note that student may still choose to abstain as one of their choices.

-End of Paper-