

SPRINGFIELD SECONDARY SCHOOL Preliminary Examination 2022 Secondary 4 Express/ 5 Normal Academic

STUDENT NAME			
CLASS		REGISTER NUMBER	

COMPUTING Paper 2 (Lab-based)

7155/02 13 September 2022 2 hours 30 minutes

Additional Materials: Electronic version of CARSALES.xlsx file Electronic version of EXCESSBAGGAGE.py file Electronic version of DICEGAME.py file Insert Quick Reference Glossary

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.

For Examiner's Use						
Task 1	/10					
Task 2	/10					
Task 3	/10					
Task 4	/20					
Total	/50					

Do not turn over this question paper until you are told to do so.

A car sales company used spreadsheet software to calculate the revenue for its sales in year 2021. The company sold 8 models of car consisting of petrol, hybrid and electric type. You are required to finish setting up the spreadsheet to calculate the revenue generated by the car sales for year 2021.

	А	В	С	D	E	F	G
1			Car	Sales for 2021			
2							
3	Car Model	Power (kW)	Car Type	Cost	Sales	Revenue (After Rebate)	Top Sales
4	A110P		Petrol	\$88,000	25		
5	B123E		Electric	\$125,000	16		
6	C150H		Hybrid	\$106,000	44		
7	D280P		Petrol	\$240,000	56		
8	E295H		Hybrid	\$255,000	23		
9	F335E		Electric	\$368,000	6		
10	G480P		Petrol	\$412,000	2		
11	H550P		Petrol	\$450,000	2		
12							
13	Total car sales						
14	Number of elect	tric car sold					
15							
16							
17	Rebat	e Table					
18	Car Type	Rebate Amount (for each car)					
19	Petrol	\$0					
20	Hybrid	\$15,000					
21	Electric	\$30,000					

Open the file CARSALES.xlsx

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

1 The three digits in the car model represents the power output of the car.

In cells **B4** to **B11** enter a formula that uses a function to show the power output of each car model. [2]

- 2 In cell **C13** enter a formula to show the total car sales. [1]
- 3 In cell C14 enter a formula to show the total number of electric cars sold. [1]
- 4 Use an appropriate function to search for the **Rebate Amount** in the **Rebate Table**, and use it to complete the **Revenue (After rebate)** column. [2]

5 In the **Top Sales** column enter a formula that uses functions to find the car model that generates the most revenue and display the text **1st** for the car model.

All other cells in the column must be empty	[2]
---	-----

6 In cells A4 to F11 use a conditional formatting tool to make the text red in the row that contains the lowest revenue. [2]

The following program is used by an airline to calculate the excess baggage cost for each passenger for the flight. Each passenger is allowed 20 kg of baggage while checking in onto a flight. They are charged \$20 per kilogram for excess baggage within the range of 20 kg to 30 kg and \$25 per kilogram for excess baggage exceeding 30 kg.

```
weight = float(input("Enter the weight of baggage: "))
if weight >= 20 and weight <= 30:
    cost = (weight - 20) * 20
elif weight > 30:
    cost = 200 + (weight - 30) * 25
else:
    cost = 0
print("The excess baggage cost is $", cost)
```

Open the file EXCESSBAGGAGE.py

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

7 The airline has a policy that disallows passengers to carry above 50 kg of baggage onboard the flight.

Edit the program to:

- reject a passenger if the baggage is above 50 kg
- allow the passenger to re-enter the weight of the baggage

Save your program.

[2]

[3]

8 The airline gives a 20% discount on the excess baggage cost for its members.

Edit the program so to:

- ask if the passenger is a member
- give a 20% discount on the excess baggage cost if the passenger is a member
- **output the message** "Thank you for being a member of ABC airline."

Save your program.

9 Edit the program so that the number of passengers for the flight is entered at the start. The total weight of the baggage and total excess baggage cost are calculated for all passengers.

Output the total weight of the baggage and the total excess baggage cost collected for the flight.

Save your program as TOTALEXCESSBAGGAGE_<your name>_<class>_<index number>.py [5]

The following program simulates two players throwing three dice to determine a winner.

Both players will be prompted to press the "Enter" key to throw the three virtual dice. A message is displayed on the screen to show the result of the throw.

The highest numbers, out of the three dice, from each player are being compared. The player with the higher number out of the comparison will win the game. There will be no winner if the highest numbers for both players are the same. A message is displayed on the screen to show the winner of the game.

The game is terminated when the player keys in "N" at the end of the two throws.

There are several syntax and logic errors in the program.

```
import math
while true:
  dice list = []
  for i in range(2):
    input("Player {}, press enter to roll the three
    dice".format(i+1))
    dice = \{0\} * 3
    for j in range(3):
      dice[j] = random.randint(0,6)
    dice list.eppend(dice)
    print("Player {}, dice = {}" .format(i,dice))
  if max(dice list[0]) > max(dice list[1]):
    print("Player 1 wins!")
  elif max(dice list[0]) = max(dice list[1]):
    print("No winners!")
  elif:
    print("Player 2 wins!")
 play = input("Play again? (Y/N): ").isupper()
  if play == 'N':
    print("Thank you for playing, goodbye!")
    break:
```

Open the file DICEGAME.py

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

10 Identify and correct the errors in the program so that it works according to the requirements given. [10]

Save your program.

A checksum formula is used to assign a last letter (suffix) to a vehicle license plate. The table below shows the suffix assigned for each checksum number.

Checksum	0	1	2	3	4	5	6	7	8	9	10	11	12
Suffix	Α	В	С	D	Е	F	G	Н	I	J	Κ	L	Μ
Checksum	13	14	15	16	17	18	19	20	21	22	23	24	25
Suffix	Ν	0	Р	Q	R	S	Т	U	V	W	Х	Y	Ζ

The vehicle license number starts with a pair of fixed alphabet "SV" followed by a variable letter and one to four digits. Example: SVA0123, SVM9918.

The steps below show how a suffix is being generated for the vehicle license plate:

• The variable letter is assigned a number based on the following:

 $A \rightarrow 0$ $B \rightarrow 1$ $M \rightarrow 12$ C = 25

• Each of the four digits are multiplied by the weight 9 and the results are sum up together with the number assigned to the variable letter.

Example of a vehicle license plate: SVA0123

Vehicle license plate	S	V	А	0	1	2	3
Assigned number / multiplication with weight	-	-	0	0 x 9	1 x 9	2 x 9	3 x 9

sum = 0 + (0x9) + (1x9) + (2x9) + (3x9) = 54

- The sum from the above step is mod divided by 26 to get the remainder which is the checksum number.
 54%26 = 2 checksum number = 2
- The assigned suffix for license plate SVA0123 will be C (from the checksum table).
- The full license plate will be SVA0123C.

11 Write a function carplate(plate), which generates a suffix for the vehicle. [9]

Save your program as **MYCARPLATE1_**<yourname>_<class>_<index number>.py

Sample executions:

```
>>> carplate("SVA0123")
C
>>> carplate("SVM9913")
V
```

12 Save your program as MYCARPLATE2_<yourname>_<class>_<index number>.py

In an updated algorithm, the digits of the license plate are multiplied by weights 4, 3, 2 and 1 with the first, second, third and fourth digits respectively. The results are summed up together with the number assigned to the variable letter.

Example of a vehicle license plate: SVZ6513

Vehicle license plate	S	V	Z	6	5	1	3
Assigned number / multiplication with weight	-	-	25	6 x 4	5 x 3	1 x 2	3 x 1

sum = 25 + (6x4) + (5x3) + (1x2) + (3x1) = 69

Extend your program to factor in the updated algorithm.

[3]

Save your program.

13 Save your program as MYCARPLATE3_<yourname>_<class>_<index number>.py

An updated version of the program allows the user to enter a car license plate with or without a suffix.

Extend your program to check if the car license plate input consists of a suffix. If it does, check if the suffix entered is correct. The program must then output the validity of the car license plate and the correct suffix. [8]

Save your program.