**BOON LAY SECONDARY SCHOOL** 



# PRELIMINARY EXAMINATION

2019

Name	
CCA	

Subject	:	COMPUTING
Paper No	:	2 (Lab-based)
Subject Code	:	7155/02
Level	:	SECONDARY FOUR EXPRESS
Date/Day	:	26 AUGUST 2019 / MONDAY
Time	:	1145 - 1415
Duration	:	2 HOURS 30 MINUTES

Additional Materials: Electronic version of TRIATHLON.xlsx Electronic version of TEMPERATURE.py Electronic version of PARITY.py Electronic version of Insert Quick Reference Glossary

## **READ THESE INSTRUCTIONS FIRST**

Before you start your exam, check that you have received the correct paper and the number of printed pages are correct.

Write your name, last four digits and the ending letter of your NRIC number, and CCA in the spaces at the top of this page.

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.

Mr. Kwek used a spreadsheet to keep track of the results in Boon Lay Secondary School's annual ironman triathlon. The triathlon consists of three segments, cycling, running and swimming, and the awards are determined by the number of points each student accumulates.

You are required to finish setting up the spreadsheet to display the information correctly.

Open the file **TRIATHLON.xlsx**. You will see the following data.

#### Save the file as

TRIATHLON\_<NAME>\_<CCA>\_<CENTRE\_NUMBER>\_<INDEX\_NUMBER >.xlsx.

	А	В	С	D	E	F	G	Н	I	J	K	L	М	N	0	Р	Q
1	Name	Gender		Swim (minutes)	Swim Points		Bike (minutes)	Bike Points		Run (minutes)	Run Points		Total Time	Special Mention		Total Points	Award
2	Carolina Chivers	F		23	1		58	2		40		1	121				
3	Graciela Greenwald	F		24	1		56	1		44			124				
4	Darla Donnellan	F		25	1		59	2		42			126				
5	Georgette Gilfillan	F		25	1		59	2		42		]	126				
6	Houston Hanel	М		30	3		58	2		50			138				
7	Cleo Carruth	F		26	1		61	3		54			141				
8	Jolyn Janke	F		33	4		58	2		38			129				
9	Jamey Janousek	М		29	2		58	2		44			131				
10	Kimbra Kamer	F		28	2		59	2		43			130				
11	Bong Brevard	М		30	3		57	1		42			129				
12	Juana Joo	F		28	2		60	2		43			131				
13	Stanford Seiler	М		31	4		56	1		47			134				
14	Harriett Hulings	F		26	1		61	3		47			134				
15	Reid Rake	F		30	3		60	2		43			133				
16	Alva Almeida	F		29	2		63	4		43			135				
17								-									
18			Average	27.8		Average	58.9		Average								
19			Fastest	23		Fastest	56		Fastest								
20			Slowest	33		Slowest	63		Slowest								
21												-					
22	Males			Swim S	core		Bike So	core		Run Sc	ore					Points	- Award
23	Females			23	1		56	1		38	1					3	Gold
24				27	2		58	2		42	2					5	Silver
25				30	3		61	3		47	3					7	Bronze
26				31	4		62	4		50	4	]				8	Nil

- 1 In cells **B22:B23** enter a formula to count the number of males and females who took part in the triathlon. [1]
- 2 In cells **J18:J20** enter a formula to calculate the **average**, **fastest** and **slowest** timings for running. The average timing must be rounded to 1 decimal place. [4]
- 3 Use an appropriate function to search for the points that each runner achieves from the **Run Score** table and use it to complete cells **K2:K16**. [2]
- Use a conditional statement to determine whether a participant will receive a special mention. For participants who have achieved a total timing not more than 125 minutes, put
   Yes in the Special Mention column, otherwise put No in the Special Mention column. [1]

- 5 In cells **P2:P16** enter a formula to calculate the total points that each participant receives from all three events. [1]
- 6 In cells Q2:Q16 enter a formula to search the **Points-Awards** table to determine the award that each participant will receive. [1]

Save and close your file.

Task 2 begins on the next page.

The following program takes an input in Fahrenheit and converts it to its corresponding temperature in degrees Celsius.

```
tempC = 0
tempF = 0
print("Temperature Conversion Program")
while True:
    tempF = float(input("Please key in a temperature in Fahrenheit: "))
    tempC = (tempF-32) * 5/9
    print(tempF, " in Fahrenheit is ", tempC, " in degrees Celsius.")
         (input("Would you like to key in another temperature?
    if
(Y/N)") =="N"):
        print("\nHave a nice day\n")
        break
Open the file TEMPERATURE.py
Save the file as:
TEMPERATURE_<NAME>_<CCA>__<CENTRE_NUMBER><INDEX_NUMBER >.py
7
      Edit the program so that it:
     (a) Outputs an empty line after "Temperature Conversion Program"
                                                                                 [1]
     (b) Calculates tempC to 1 decimal place.
                                                                                 [2]
     (c) Prints out the message "This temperature is in the acceptable range"
          when tempC is calculated to be between 40.1°C and 44.9°C inclusive.
                                                                                 [5]
      Save your program.
8
      Save your program as:
```

ANY\_TEMPERATURE\_<NAME>\_<CCA>\_\_<CENTRE\_NUMBER><INDEX\_NUMBER
>.py

[3]

Edit your program so that it allows the user to specify how many temperature readings he/she would like to input.

Save your program.

A parity bit is a simple form of validation for detecting errors in data transmitted over a network.

In even parity, the total number of 1s in each group of 8 bits must be an even number.

In odd parity, the total number of 1s in each group of 8 bits must be an odd number.

The program uses the following algorithm:

- Prompts the user to decide between even or odd parity
- Prompts the user to key in 8 bits
- Counts and outputs the number of 1s in the input
- Outputs the corresponding parity bit depending on whether even or odd parity was chosen at the beginning

There are several syntax and logic errors in the program.

```
print("Parity checker")
parity = '
byte = ''
count1 = 1
while True:
    parity = int(input("Which parity would you like to use? (Even/Odd):
"))
    if parity == 'Even' and parity == 'Odd':
        break
    else:
        print("Please key in 'Even' or 'Odd'.")
while True:
    byte = input("Please key in the 8 bits: ")
    for bit in byte
        if bits == '1':
            count1 == 1
    print("There are ", count1, "1s in ", byte)
    if parity == 'Even':
        if count1 // 2 == 0:
            print("Parity bit == 1")
        else:
            print("Parity bit == 1")
    else:
        if count1\%2 != 0:
            print("Parity bit == 1")
        else:
            print("Parity bit == 0")
```

#### Open the file PARITY.py

Save the file as **PARITY\_**<NAME>\_<CCA>\_\_<CENTRE\_NUMBER><INDEX\_NUMBER >.py

**9** Identify and correct the errors in the program so that it works according to the rules given.

Save your program.

[10]

Task 4 begins on the next page.

"Rock, Paper, Scissors" is a game that involves two players. The players use their hands to make a representation of either rock, paper or scissors simultaneously and the winner is determined by a set of rules.

You have been tasked to write a program to allow two players to play this game on a computer.

The program should:

- Output the title of the game: "Rock, Paper, Scissors!"
- Allow player 1 to input either Rock, Paper or Scissors. There must be validation present to check that only valid inputs have been keyed in.
- Allow player 2 to input either Rock, Paper or Scissors.
- Determine the winner of the round using the following rules:
  - $\circ \quad \text{Rock wins scissors}$
  - o Scissors wins paper
  - o Paper wins rock
- Outputs the winner of each round
- **10** Write your program and test that it works.

Save your program as: **RSP\_**<NAME>\_<CCA>\_\_<CENTRE\_NUMBER><INDEX\_NUMBER >.**py** [10]

- 11 When your program is complete, test it for the following:
  - Test 1 Player 1 keys in 'rock'
  - Test 2 Player 1 keys in Rock and Player 2 keys in Paper
  - Test 3 Player 1 keys in Paper and Player 2 keys in Scissors

Take a screen shot of:

- Test 1 and 2. Save this screenshot as: TEST12\_<NAME>\_<CCA>\_\_<CENTRE\_NUMBER><INDEX\_NUMBER >
- Test 3. Save this screenshot as: TEST3\_<NAME>\_<CCA>\_\_<CENTRE\_NUMBER><INDEX\_NUMBER >.

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

[3]

12 Modify your code to allow the players to determine how many rounds they would like to play.

Save your program as

**ROUNDS\_RSP\_**<NAME>\_<CCA>\_\_<CENTRE\_NUMBER><INDEX\_NUMBER >.py

[2]

- **13** Extend your code in **Q12** to:
  - Track and output the score at the end of each round
  - Output the winner after all the rounds have been completed.

Save your program as

TRACK\_RSP\_<NAME>\_<CCA>\_\_<CENTRE\_NUMBER><INDEX\_NUMBER >.py

[3]