

ZHONGHUA SECONDARY SCHOOL PRELIMINARY EXAMINATION 2018 SECONDARY 4 EXPRESS

Candidate's Name	Class	Register Number

COMPUTING

Paper 2

7155/02 29 August 2018 2 hours 30 minutes

Additional Materials:

Data Resource Files Quick Reference for Python sheet

READ THESE INSTRUCTIONS FIRST

Write your index number and name on all the work you hand in. Write in dark blue or black pen on both sides of the paper. You may use a pencil for any diagrams or graphs. Do not use staples, paper clips, glue or correction fluid.

Answer all the questions.

Write your answers in this question booklet where required. All other answers in the form of software files must be saved on the provided thumb drive, within the stated file folders.

You are reminded of the need for clear presentation in your answers.

At the end of the examination, fasten all your work securely together. 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:	

Setter: Mr. Calvin Heng Vetter: Mr. Low Kee Ley

A Book Reading club uses spreadsheet software to record the money received and how many years the member has been with the club. You are required to finish setting up the spreadsheet to record the cashback entitlement that will be paid.

Open the file **MEMBERSAVINGS**. You will find the following data.

Save the file as MEMBER_<YOUR NAME>_<INDEX NO>

	A	в	С	D	E	F
1	XYZ Club Membership Financials 2018					
2						
3	1ember Numbe	Annual Subscript ion	Number of Years as Member	Special Discount Bate	Cashback Entitlement	Free Gift (Pink or Green
4	GB001	\$200.00	2			
5	GB002	\$200.00	3			
6	GB003	\$250.00	7			
7	GB004	\$250.00	8			
8	GB005	\$500.00	8			
э	GB006	\$750.00	5			
10	GB007	\$750.00	4			
11	GB008	\$250.00	2			
12	GB009	\$200.00	1			
13	GB010	\$500.00	6			
14	GB011	\$250.00	6			
15	GB012	\$200.00	2			
16						
17	Total					
18	umber of Members					
19	Range of Years	Range of Years		Discount Rates		
				Number of		Special
				Years as	Description	Discount
20				Member		Rate
21				1	One year	1%
22				2	Two or Three years	2%
23				4	Four or Five years	3%
24				6	Six years and above	4%
25						

- 1 In cell **B17** enter a formula to calculate the total amount of [1] annual subscriptions.
- 2 In cell **B18** enter a formula to count the number of members of [1] the book reading club.
- 3 In cell **B19** enter a formula to calculate the range of number of [1] years as member.
- 4 In cell **D4**, use an appropriate function to search for the **Special** [2] **Discount Rate** in the **Discount Rates** table and use it to complete the **Special Discount Rate** column.

5 In cell **E4**, enter a formula to calculate the Cashback Entitlement [2] for members and use it to complete the **Cashback Entitlement column**. Round these values to the nearest whole number.

(hint: Cashback Entitlement = Annual Subscription x Special Discount Rate)

6 In cell **F4**, use a conditional statement, to identify those [2] members who have contributed \$750 with more than three years membership and put **PINK** in the **Free Gift** column. Otherwise put **GREEN** in the **Free Gift** column.

Save your file.

Open the tab **FINANCIAL**. You will see the following data.

	А	В	С	D	E	F	G
1	Task 1	Question 7					
2							
3	Anthony wishes to take a study loan at an interest rate of 3.2% per annum						
4	compounded monthly that can be paid back with a lump sum of \$70,000 after 8						
5	years. Using an appropriate function, find the maximum amount Anthony can						
6	afford.						
7							
8	The maximum amount he can afford is:						

7 Anthony wishes to take a study loan at an interest rate of 3.2% [1] per annum compounded monthly that can be paid back with a lump sum of \$70,000 after 8 years. Using an appropriate function, find the maximum amount Anthony can afford. Enter your answer in cell **E8**.

Save and close your file.

The following program accepts timings (mm:ss) for the 2400m running event, one for each completion of a 400m circuit, for a total of 6 values. It will print out the average timing and the lowest timing. Timings are in the range 00:00 to 30:00.

Open the file RUNTIMINGS.py

Save the file as RUNNING_EVENT _<YOUR NAME>_<INDEX NO>

- 8 Edit the program so that it:
 - (a) Accepts 12 timing entries, as the organizer wants to take [1] timings at 200m intervals.
 - (b) Print out the highest timing entry as well as the average [4] timing entry.
 - (c) Test if the timing entry input is between 00:00 and 30:00, if [3] not, ask the user for input again as necessary.

Save your program.

- 9 Save your program as VAR_INTERVALS_<YOUR NAME>_<INDEX NO>
- 10 Edit your program so that is works for any number of regular [2] intervals.

Save your program.

The following program converts a range of Fahrenheit temperature readings to Celsius and vice versa. It begins by allowing the user to choose between an "F" for Fahrenheit to Celsius conversion or "C" for Celsius to Fahrenheit conversion.

The formula for converting Fahrenheit to Celsius is: $C = 5/9 \times (F - 32)$

The formula for converting Celsius to Fahrenheit is: F = 32 + (C * 9/5)

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

```
100 def displayWelcome():
110
       print("This program will convert a range of temperatures")
120
       print("Enter (F) to convert Fahrenheit to Celsius")
130
       print("Enter (C) to convert Celsius to Fahrenheit\n")
140
150 def getConvertTo():
       which = input("Enter selection: ")
160
       while which == "F" or which == "c":
170
180
        which = input("Enter selection: ")
190
       return which
200
210 def displayFahrenToCelsius(start, end):
220
       print("\n Degrees", " Degrees")
       print("Fahrenheit", "Celsius")
230
240
250
       for temp in range(start, end + 1):
260
          converted temp = temp - 32 * 5/9
          print(" ", format(temp, "4.1f"), " ", format(converted temp,
270
"4.1f"))
280
290 def displayCelsiusToFahren(start, end):
       print("\n Degrees", "Degrees")
300
310
       print(" Celsius", "Fahrenheit")
320
330
       for temp in range(start, end):
340
          converted temp = (9/5 * \text{temp}) + 32
          print(" ", format(temp, "4.1f"), " ", format(converted temp,
350
"4.1f"))
360
370 # --- main
380
390 #Display program welcome
400 displayWelcome()
410
420 # Get which conversion from user
430 which = getConvertTo()
```

440

- 450 # Get range of temperatures to convert
- 460 temp_start = int(input("Enter starting temperature to convert: "))
- 470 temp_end = input("Enter ending temperature to convert: ")
- 480
- 490 # Display range of converted temperatures
- 500 if which == "F":
- 510 displayFahrenToCelsius(temp_start, temp_end)
- 520 elif which == "c":
- 530 displayCelsiusToFahren(temp_start, temp_end)

Open the file **TEMPCONV_BUGS.py**

Save the file as TEMPCONV_DEBUG_<YOUR NAME>_<INDEX NO>

12 Identify and correct the errors in the program so that it works correctly [10] according to the rules above.

Save your program.

In the Kingdom of Octavia, there exists a practice where suitable candidates are vetted for service to the King. Part of the vetting requires you, the Grand Statistician, to provide a report on a study of candidate names. To this end, you have created a process to create this report so that the Special Civilian Selection Committee can carry out further analysis on the candidates.

- **13** Create a program that allows the input of 10 names into a [1] suitable data structure.
- 14 On inputting 10 names into the data structure, you discover that [1] data entry can sometimes lead to a mix of uppercase and lowercase letters entered in a name. Hence, your program should find a way to ensure that names in the data structure are stored as uppercase letters.
- **15** Sort the names in alphabetic order. [1]
- 16 Calculate the name_worth for each name. Using the **ORD()** [3] function or otherwise, the number worth of each name is defined as the sum of each letter, converted to its numerical ASCII equivalent, after that, the value is multiplied by the position of the name in the data structure. Please note that the first name in the data structure is in position 1.
- **17** For the data structure of 10 names, calculate the score. The [2] score is the addition of all the name worth values.
- **18** Produce the following report:

Name	Worth
 XXXXXXX YYYYYYY	9999 99999
 ZZZZZZZZZ	9999
Score	999999

19 Test your program using the following test data: [2]

MARY Jane Lester WILLIAM calvIN Henry CHArles JOLIET COULSON edison [2]

20 Take a screen shot of your report. Name your bitmap file as: [1]

WORTHY_SCREENSHOT_<YOUR NAME>_<INDEX NO>

21 Save your program as:

WORTHY_ORDPROG_<YOUR NAME>_<INDEX NO>

- 22 You are required to enhance your program with the following requirements:
 - (a) A different way of calculating the name_worth is as follows:
 - (i) if the character is a vowel, the value calculates as: [3]

numerical value * square root of 2

(ii) if the character is a consonant, the value calculates as:

numerical value * 3 * pi * square root of 3

hint: use of the python built-in math library is encouraged.

- (b) Your program is to allow for any number of names to be [2] entered.
- 23 Save the enhanced program as:

WORTHY_ENHPROG_<YOUR NAME>_<INDEX NO>

24 You are strongly encouraged to demonstrate good [2] programming and documentation practices.

----- End of Paper -----