PATHLIGHT SCHOOL
PRELIMINARY EXAMINATION
SECONDARY 4 EXPRESS

CANDIDATE NAME								
CENTRE NUMBER	S					INDEX NUMBER		

COMPUTING 7155/02

Paper 2 Practical (Lab-based) **August 2022**

2 hours 30 minutes

Additional Materials: Electronic version of `CLUB.xlsx` data file
Electronic version of `CHECKAGE.py` file
Electronic version of `LEAPYEAR.py` file
Insert Quick Reference Glossary

READ THESE INSTRUCTIONS FIRST

Write your Centre number, Index number and name 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 marks for this paper is 50.

Task 1

A fitness club uses spreadsheet to keep track of its members' particulars and activities.

As part of its loyalty programme, the club is planning the following benefits for its members this year:

- membership discount based on a tiered discount programme
- members who have joined for 3 years and above and have been active will be given a special gift set

You are required to finish setting up the spreadsheet for the fitness club.

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

	A	B	C	D	E	F	G	H	I	J
1	**Date**									
2	**Member name**	**Date of Birth**	**Member ID**	**Number of Years Joined**	**Active?**	**Fees Outstanding**	**Discount**	**Fees Payable**	**Gift Set?**	
3	Josephine Teo	12/10/1979	JOS792012		Yes	540				
4	Joshua Chong	23/2/1972	JOS722010		Yes	540				
5	Jason Tan	25/7/1988	JAS882020		No	1080				
6	Ashley Tang	6/5/2000	ASH002022		Yes	1080				
7	Alexis Chin	4/9/1990	ALE902015		Yes	540				
8	Hailey Tang	6/5/2000	HAI002022		Yes	540				
9	Robert Ng	1/4/1993	ROB932020		Yes	540				
10	Claire Carter	16/10/1998	CLA982018		Yes	540				
11	How Lai Ling	18/7/1999	HOW992018		No	1080				
12	Jeremy Yeo	30/12/2002	JER022020		No	1080				
13	Tan Eng Chye	3/6/1965	TAN652011		No	1620				
14	Melinda Tay	9/8/1980	MEL802019		Yes	540				
15										
16	**Membership Discount**									
17	**Number of Years**	**Tier**	**Discount**							
18	0	1	0%							
19	5	2	5%							
20	10	3	10%							
21										

Save the file as **MYCLUB**_<your name>_<centre number>_<index number>**.xlsx**

1 In cell **B1** enter a function to show today's date. [1]

2 The last four characters of the **Member ID** shows the year the member joined the club.

In cells **D3** to **D14**, enter a formula to find the number of years that each member has joined the club. [2]

[Hint: the function =YEAR() extracts the year from a date.]

3 In cells **G3** to **G14** enter a formula that uses an appropriate function to search for the **Discount** in the **Membership Discount** table to calculate the discount given for each member. [2]

4 In cells **H3** to **H14** enter a formula to calculate the **Fees Payable** for each [1]

member.

5 Use a conditional statement to identify whether a member will receive a gift set. For members who have joined the club for 3 years or more and have been active, put **Yes** in the **Gift Set** column, otherwise put **No** in the **Gift Set** column. [2]

6 In cells **A3** to **I14** use a conditional formatting tool to change the background colour of the member name to yellow in the row that contains the youngest member. [2]

Task 2

The following program checks whether a person is able to join a fitness club.

The club does not recommend children joining the club before they are 14 years old.

The program requires the input of the names and age of 10 people. It will display an appropriate message after each person's details are input, to indicate if the person is old enough to join the club as a member or not.

```
member_no = 10

minimum_age = 14

for i in range(member_no):
    name = input("Name of person: ")
    age = int(input("Age of person: "))

    if age >= minimum_age:
        print("The person is old enough.")
    else:
        print("The person is not old enough.")
```

Open the file **CHECKAGE.py**

Save the file as **MYCHECKAGE**_<your name>_<centre number>_<index number>**.py**

7 Edit the program so that it will work for any number of people. The program must display a suitable input message.

Save your program. [1]

8 A person must be fit enough to attend the high intensity training programme in the club. A person is deemed unfit if he or she has a medical condition.

Edit the program so that it checks that a person is old enough **and** fit enough to attend the high intensity training programme. The program must display:

- "`The person is not fit enough.`", if a person is old enough but not fit enough
- "`The person is not old enough.`", if a person is fit enough but not old enough
- "`The person is not fit enough and not old enough.`", if a person is both not fit enough and not old enough
- "`The person is old enough and fit enough.`", if a person is old enough and fit enough

Save your program. [5]

9 Edit the program so that it stores the name of the person in a list, only if the person is old enough and fit enough. Each name must be stored in the next available element in the list.

Output the list after all people have been entered.

Save your program as

CHECKAGEFITNESS_<your name>_<centre number>_<index number>**.py** [4]

Task 3

Most years have 365 days. However, the time required for the Earth to orbit the Sun is actually slightly more than that. As a result, an extra day, February 29, is included in some years to correct for this difference. Such years are referred to as leap years.

The rules for determining whether or not a year is a leap year follow:

- Any year that is divisible by 400 is a leap year.
- Of the remaining years, any year that is divisible by 100 is **not** a leap year.
- Of the remaining years, any year that is divisible by 4 is a leap year.
- All other years are **not** leap years.

The program below reads 10 inputs of years from the user and displays a message indicating whether or not each input year is a leap year.

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

```
count = 10

while count < 10:
    year = int(input("Enter a year: ")

    if year % 400 = 0:
        isLeapYear = True
    elif year % 100 == 0:
        isLeapYear = True
    elif year // 4 == 0:
        isLeapYear = True
    elif:
        isLeapYear = False

if isLeapYear:
    print(year, "is not a leap year.")
else:
    print(year, "is a leap year.)

count = count - 1
```

Open the file **LEAPYEAR.py**

Save the file as **MYLEAPYEAR**_<your name>_<centre number>_<index number>**.py**

10 Identify **and** correct the errors in the program so that it works correctly accordingly to the rules given. [10]

Save your program.

Task 4

A parity bit is a simple mechanism for detecting errors in data transmitted over an unreliable connection such as a telephone line. The basic idea is that an additional bit is transmitted after each group of 8 bits so that a single bit error in the transmission can be detected.

The following describes how parity bits work:

- Parity bits can be computed for either even parity or odd parity.
- If even parity is selected then the parity bit that is transmitted is chosen so that the total number of one bits transmitted (8 bits of data plus the parity bit) is even.
- When odd parity is selected the parity bit is chosen so that the total number of one bits transmitted is odd.

You have been asked to write a program to compute the parity bit of an 8-bit binary number.

The program must:

- use a user-defined function to compute the parity bit for groups of 8 bits entered by the user using even parity.
- allow a user to input strings containing 8 bits until the user enters a blank line (i.e. empty input)
- display a clear message indicating whether the parity bit should be 0 or 1
- display an appropriate error message if the user enters something other than 8 bits.

[Hint: You should read the input from the user as a string. Then you can use the `<str>.count` method to help you determine the number of ones in the string or write a function to do it.]

11 Write the program to meet the requirements. Save your program as
EVENPARITY_<your name>_<centre number>_<index number>**.py** [12]

12 When your program is working, use the following test data to show your test result.

- Test 1 – User inputs `01010011`
- Test 2 – User inputs `1100000` followed by `11000001`
- The user then enters a blank line to end the program

Take a screenshot of the results of Test 1 and Test 2. Save this **single** screenshot as:

PARITYTEST_<your name>_<centre number>_<index number> [3]

Save your file in either **.png** or **.jpg** format.

13 Save your program as:
CHOOSEPARITY_<your name>_<centre number>_<index number>**.py**

Extend your program to:

- Allow user to choose even or odd parity.
- Include a function to compute the parity bit using odd parity. [5]

Save your program.

END OF PAPER

BLANK PAGE

BLANK PAGE