Name: _____ **Class:** _____ **Index Number:** _____

Anglo-Chinese School
(Barker Road)

PRELIMINARY EXAMINATION 2021

SECONDARY FOUR
EXPRESS

COMPUTING
PAPER 2

7155/02

2 HOUR 30 MINUTES

INSTRUCTIONS TO CANDIDATES

Additional Materials: Electronic version of TICKETS.XLSX data file
Electronic version of LOGIN.PY Python file
Electronic version of RESULTS.PY Python file
Insert Quick Reference Glossary

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, inclusive of the cover page*

Task 1

XYZ Airlines uses spreadsheet software to calculate the sales of air tickets to ABC island. Customers can choose between first, business or economy class for their air tickets. They are also entitled to additional discounts according to membership status.

You are required to finish setting up the spreadsheet to calculate the cost of each transaction.

Open the file **TICKETS.xlsx**. You will see the following data. Only selected rows from the file have been shown below.

	A	B	C	D	E	F	G	H
1	**XYZ AIRLINE TICKETING SALES TO ABC ISLAND**							
2	**Name of Customer**	**Cabin Type**	**Number of Tickets**	**Membership**	**Early Bird**	**Tickets Cost**	**Discount**	**Final Cost**
3	Lesha Tedeschi	First	2	Premier	Yes			
4	Emanuel Riemer	Economy	4	No	No			
5	Brittanie Silvernail	First	1	No	No			
6	Julianne Eyler	Business	5	Basic	No			
7	Aracelis Manthey	Business	2	No	No			
21	Diana Rowley	Economy	8	Basic	No			
22	Laila Batey	Business	1	No	No			
23	Karima Molder	First	2	Premier	Yes			
24	Ossie Luedtke	Economy	3	Basic	Yes			
25	Tang Xinyi	Economy	5	Premier	No			
26								
27	**Ticket Information**							
28	**Cabin Type**	**Price**	**Number Sold**					
29	First	$1,000						
30	Business	$500						
31	Economy	$200						
32								
33	**Membership Type**	Premier	Basic	No				
34	**Discount rate**	20%	10%	0%				

Save the file as **MYTICKETS**_<your name>_<centre number>_<index number>

1 The **Tickets Cost** needs to be calculated for the correct **Number of Tickets** the customer has ordered.

In cells **F3** to **F25** enter a formula that uses an appropriate function to search for the **Price** of each ticket in the **Ticket Information** table. Use it to calculate the **Tickets Cost** of each transaction. [2]

2 The **Discount** can be calculated by applying the appropriate **Discount rate** depending on the **Membership** of the customer.

In cells **G3** to **G25** enter a formula that uses an appropriate function to search for the **Discount rate** of each transaction in the **Membership Type** table. Use it to calculate the **Discount** of each transaction.. [2]

3 In cells **H3** to **H25** enter a conditional statement to calculate the **Final Cost** of each transaction. This is the **Tickets Cost** subtracted by the **Discount**. An additional 5% reduction is applied to the **Final Cost** for **Early Bird** transactions. [2]

4 In cells **C29**, **C30** and **C31** enter a formula that uses an appropriate function to calculate the number of tickets sold for each cabin type. [2]

5 In cells **A3** to **H25** use a formatting tool to change the colour of the row to blue if the Cabin Type is First. [2]

Save and close your file.

Task 2

The following program creates a username for a user. It creates the username by taking the first letter of the user's first name and combining it with the user's last name. It will also allow the user to enter a personal identification number (PIN).

```
firstname = input("Please enter your first name: ")
lastname = input("Please enter your last name: ")
username = firstname[0] + lastname
print("Your username is {}".format(username))
pin = int(input("Please enter a PIN: "))
```

Open the file **LOGIN.py**

Save the file as **MYLOGIN**_<your name>_<index number>

6 Edit the program so that the username is created using the first three letters of the first name along with the last three letters of the last name. The username should be in uppercase. [3]

Save your program.

7 Edit the program to:

(a) only accept a PIN of six digits to be entered by the user. A suitable error message must be displayed if an incorrect input is given. The program must loop until the PIN input is six digits. [2]

(b) ask the user to re-enter the PIN. The program will check that the second entry of the PIN matches the first entry. It will output the message "Your PIN has been set." if the PIN entries match. Otherwise, output the message "PIN entries do not match. Please repeat." and read in the first PIN entry again, and repeat this until the PIN entries match. [3]

Save your program.

8 The program also needs to ensure that both first name and last name entered are words with three or more alphabets.

Edit your program to test whether the first name is a word of three alphabets or more with no spacing in between. Output a suitable error message and quit the program if the first name does not meet the above requirements. [2]

There is no need to validate the last name.

Save your program.

Task 3

The following program collates the names and scores of students in a class.

The program validates that the score is from 0 to 100 inclusive, and also computes and outputs:

- number of scores entered
- number of students who score distinction (75 or more)
- number of students who failed (below 50)
- average score in the class

The program allows a user to continue entering the names and scores until the user enters the letter 'N' when prompted.

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

```
name_list = []
mark_list = []
dist_list = []
pass_list = []
fail_list = []
count = 1

flag = True
while flag == False:
    name = input('Enter student's name: ')
    name_list += [name]
    while True:
        mark = int(input('Enter score of student: '))
        if mark >= 0 or mark <= 100:
            break
        else:
            print('Invalid mark!')
        mark_list += [mark]
    count += 1
    if mark > 75:
        dist_list += [name]
    elif mark >= 50:
        pass_list += [name]
    else:
        fail_list += (name)
    more = int(input('Would you like to enter another score, Y or N?: '))
    if more == 'N':
        flag = False

average = round(max(mark_list)/len(mark_list), 2)
num_dist = len(dist_list)
num_fail = len(fail_list)

print("You entered " + count + " scores.")
print(str(num_dist) + " students score distinction and " + str(num_fail)
      + " students failed.")
print("Average score is " + str(average))
```

Open the file **RESULTS.py**

Save the file as **MYRESULTS**_<your name>_<index number>

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

Save your program.

Task 4

You have been asked to create an attendance program for a school.

The program should:

- Enter the data for five students in the format *p q r s t* where p, q, r, s, t are the attendance status of each student for the day. An example is `P A P P E`
- Only allow data entry of `P` for present, `E` for Excused or `A` for Absent. Output "`Invalid. Try again.`" and prompt for retry if data entry is not `P`, `E` or `A`.
- Repeat this for a total of five days from Mon to Fri
- Calculate the number of students present for each day
- Calculate the weekly attendance rate for the whole week rounded to 1 decimal place. [Weekly attendance rate = number of present students in the week / total possible attendance in the week * 100%]
- Display this on the screen. Your output **must** look like this:

```
Enter attendance for Mon: P P P A E
Enter attendance for Tue: P E E P A
Enter attendance for Wed: P P P P P
Enter attendance for Thu: P P E P P
Enter attendance for Fri: P E P P P

Mon  3 student(s) are present
Tue  2 student(s) are present
Wed  5 student(s) are present
Thu  4 student(s) are present
Fri  4 student(s) are present

Weekly attendance rate = 72.0%
```

10 Write your program and test that it works.

Save your program as **ATT1**_<your name>_<index number>**.py** [12]

11 When your program is complete, use the following test data to show your test results:

```
P X P A P
P P P A E
P E E P A
P P P P P
P P E P
P P E P P
P E P P P
```

Take a screen shot of your results and save it as a bitmap **ATTRESULTS**_<your name>_<index number>

[3]

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

12 Save your program as **ATT2**_<your name>_<index number>**.py**

Extend your program to identify students that have 4 or more days of attendance in a week. Print out the student number and the number of days he is present. Your output should look like this.

```
Student 1 is present for 5 days.
Student 4 is present for 4 days.
```
[3]

Save your program.

13 Save your program as **ATT3**_<your name>_<index number>.**py**

Extend your program to work for any number of students. [2]

Save your program.

End of Paper