

JUNYUAN SECONDARY SCHOOL **PRELIMINARY EXAMINATION 2022** SECONDARY FOUR EXPRESS

CANDIDATE NAM	Ξ	
CLASS		
COMPUTING		7155/02
Paper 2 (Practical)		12 Sep 2022
		2 hours 30 minutes
Additional Materials:	1 x Thumb Drive Electronic version of TASK1.xlsx file Electronic version of TASK2.py file Electronic version of TASK3.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		

Ms. Yeo is consolidating the academic results of her students at the end of the year.

She uses a spreadsheet software to make calculations and plan for school events. Students who do well will be awarded a monetary award on Speech Day.

Ms. Yeo also plans to help her students who are underperforming by conducting a holiday bridging programme.

You are required to finish setting up the spreadsheet to record the details for her students.

Open the file **TASK1.xlsx** and you will see the following data.

	A B	С	D	Е	F	G	Н	I.	J	К	L	М
1	Mark Sheets and Grade Summary For Year 2028											
2												
3	S/N Name	ENG	MATH	SCI	Average	Award	Top Student?					
4	1 ALICIA CHUA WAN YI	73	68	52								
5	2 AMANDA CHIN HUI YIE	74	85	62								
6	3 CLOEY KANG YU HUA	68	70	70								
7	4 IKA SYAHIDAH HIDAYAH BTE ATAN	81	77	33								
8	5 KHOO QI QUAN	69	85	32					Speech Day Awards			
9	6 LAU WEI YUAN	70	39	63					Average	Sponsor	Award	Amount
10	7 LIEW YU CHUN	72	47	63					0	-	-	0
11	8 NADZIRAH BINTE JASMAN	65	80	47					60	PSG	Bursary	350
12	9 NASYA AULIA AZZAHRA	65	88	54					70	SAB	Scholarship	1000
13	10 NUR ALIYAH YUSRINA BINTE MOHAMAD YASIR	77	78	29								
14	11 NUR ALYSYA NATASYA BINTE SHAHARUDIN	73	68	70								
15	12 NUR DIYANAH BINTE ABDUL HAMMED	68	41	54					Holiday Brid	ging Progra	mme	
16	13 NUR JAZLINA BINTE ZUKIPILI	67	71	71					Group	Number	Criteri	a
17	14 NUR KAMILIA BINTE MASKAM	64	76	56					Α		55 < Average <=	40
18	15 NUR SYAFIQAH BINTE ISHAK	75	85	57					В		Average < 40	
19	16 NUR SYARAFANA BINTE SA'AT	67	91	61								
20	17 NURYN BINTE ZAINAL	66	38	45								
21	18 PUTRI RAHADATUL 'AISY BINTE MOHAMMAD R.	67	83	41								
22	19 RABI'ATUL A'DAWIYAH BINTE MOHAMAD SHAV	77	90	48								
23	20 SITI HAJAR BINTE ABD SHUKOR	68	71	36								
24	21 SOFIA KALLYSTA BINTE NOOR EFFENDY	75	84	36								
25	22 TANG YEE TENG RANICE	66	70	72								
26	23 WENG EE LYN	67	64	61								
27	24 YASMIN HAZMIRA BINTE AMIR	70	23	36								

Save the file as **MARKS**_<Class>_<Index>_<Name>.xlsx.

- 1 In the **Average** column, enter a formula that uses a function to calculate the average of all the subjects attained for each student rounded up to one decimal place. [2]
- In the Award column, enter a formula that uses an appropriate function to search for the Award in the Speech Day Awards table. [2]

3 A student is awarded a Top Student award if their average score is the highest.

In the **Top Student** column, enter a formula that uses functions to find the student(s) with the highest average and display the text 'Top' for the student.

All other cells in the column must be empty.

4 In cells K17 and K18, enter a formula that uses an appropriate function to count the number of students that meets the criteria description in the Holiday Bridging Programme table. [3]

Save your work.

[3]

The following program accepts positive integers and outputs their corresponding binary (base-2) equivalent.

```
a = int(input("Enter a denary number: "))
converted = 0
loop = 0
while a != 0:
   remainder = a % 2
   converted = converted + remainder*(10**loop)
   a = a // 2
   loop += 1
```

print(converted)

Open the file **TASK2**.py

Save the file as BINARY_<Class>_<Index>_<Name>.py

5 Edit the program so that when a denary number 10 is entered, it outputs the following statement: Binary equivalent of 10 is 1010.

6 Edit the program to accept and convert only five positive integers from the user.

Save your program.

Save your program.

7 Edit the program to only accept positive integers as input. A suitable error message must be displayed if a negative input is given. The program must loop until a valid denary number is input.

Save your program.	[3]

8 Save the file as **OCTAL**_<Class>_<Index>_<Name>.py

Edit the program so that it converts the positive denary numbers into octal numbers. Similar to the binary number system, octal numbers are base-8 numbers.

Save your program.

[2]

[2]

[3]

A program converts a denary number to its hexadecimal equivalent and vice versa.

The program shown on the next page asks the user to input a conversion option:

- H to convert denary numbers to hexadecimal numbers
- D to convert hexadecimal numbers to denary numbers
- Q to end the program.

The conversions are processed in subfunctions.

To convert from denary number to hexadecimal number, the function:

- takes in the user input
- divides the denary number repeatedly by 16
- uses the remainders of each division to derive the hexadecimal number from the hexadecimal array
- divides the quotients of each division until the quotient is zero.
- returns the hexadecimal number

To convert the hexadecimal number to denary number, the function:

- takes in the user input
- flips the hexadecimal input
- locates the index of each character in the input in the hexadecimal array
- computes the denary number by multiplying the hexadecimal value with its place value
- returns the denary number

The program outputs the conversion result. The program loops the options until the user enters a 'Q' or 'q' to end the program.

There are several syntax and logic errors in the program.

```
def hextoden(a):
   converted = 0
   hex sys = ["0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "A", "B",
"C", "D", "E", "F"]
   a = a[::-1]
    for i in range(len(a)):
        value = hex sys.index(a[i])
        converted = converted + value*(16*i)
    return converted
def dentohex():
    converted = ""
   hex sys = ["1", "2", "3", "4", "5", "6", "7", "8", "9", "A", "B", "C",
"D", "E", "F"]
   while a != 0:
       remainder = a % 8
       converted = hex sys[remainder] + converted
       a = a / 16
    return converted
choice = input("Enter H to convert denary to hex \n
Enter D to convert hex to denaryn
Enter Q to Quit: ").upper()
while choice != 'Q'
   user = input("\nEnter the number to be converted: ")
    if choice == 'H':
        hex num = dentohex(user)
        print("Hexadecimal of", user, "is", hex num, ".")
    if choice != 'D':
        den num = hextoden(user.upper())
        print("Denary of", user.upper(), "is", den num, ".")
    choice = input("\nEnter H, D or Q: ").upper()
print("Program ends."
```

Open the file TASK3.py

Save the file as **HEXDEN_**<Class>_<Index>_<Name>.py

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

Save your program.

[10]

Ms. Yeo wants a program that collates the names and scores of her students in a class.

The program must:

- ask the user for the number of student data that is being entered
- ask the user to enter a student name, converts the input to upper case and store the input
- ask the user to enter the score of the student, correct to one decimal place and store the input
- repeatedly take input of a student's score until the score is from 0 to 100 inclusive
- tracks the names of students who achieved distinction (75 marks or more)
- tracks the names of students who failed (below 50 marks)
- computes and outputs the following:
 - number of students who achieved distinction and failed
 - the names of the students who achieved distinction, if any
 - the names of the students who failed, if any
 - average score of the class, correct to one decimal place

Your program must include appropriate input and output messages.

10 Write a program to meet the requirements.

Save your program as **CLASS1_**<Class>_<Index>_<Name>.**py** [15]

11 When your program is working, use the following test data of five students to show your result.

Student Name	Score
Mike Wheeler	65.9
Lucas Sinclair	-9
	70.3
Max Mayfield	91
Will Byers	61.6
Dustin Henderson	82.4

Take a screenshot of your result and save it as:

TESTCLASS_<Class>_<Index>_<Name>

[2]

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

11 Save your program as CLASS2_<Class>_<Index>_<Name>.py

The classes are named by level in alphabetical order. There are 4 levels (Level 1, 2, 3, 4) and each level has 5 classes. For example, Level 2 classes are named 2A, 2B, 2C, 2D and 2E.

Extend your program to:

- ask the user to enter the class of the students
- repeatedly take input of the class until it is valid
- outputs the class before other data

Save your program.

[3]

End of Paper