Sec 4 Computing 2022 Prelim Paper 2
Marking Scheme

Task 1:

	A	B	C	D	E	F	G
1	2020 Application for Covid-19 Crisis Fund						
2							
3							
4	**Application Ref. No.**	**Age**	**Citizenship**	**Monthly Family Income**	**Number of Family Members**	**Monthly Per Capita Income**	**Cash Payout**
5	C1950893	=2020-MID(A5,2,4)	=IF(LEFT(A5,1) = "C","Citizen", IF(LEFT(A5,1) = "F", "Foreigner", "Permanent Resident"))	3200	6	=FLOOR(D5/E5,1)	=IF(OR(C5="Citizen",C5="Permanent Resident"),VLOOKUP(F5,$E$29:$G$31,3,TRUE),0)
6	C1959252	=2020-MID(A6,2,4)	=IF(LEFT(A6,1) = "C","Citizen", IF(LEFT(A6,1) = "F", "Foreigner", "Permanent Resident"))	1800	3	=FLOOR(D6/E6,1)	=IF(OR(C6="Citizen",C6="Permanent Resident"),VLOOKUP(F6,$E$29:$G$31,3,TRUE),0)
7	P1971158	=2020-MID(A7,2,4)	=IF(LEFT(A7,1) = "C","Citizen", IF(LEFT(A7,1) = "F", "Foreigner", "Permanent Resident"))	4500	3	=FLOOR(D7/E7,1)	=IF(OR(C7="Citizen",C7="Permanent Resident"),VLOOKUP(F7,$E$29:$G$31,3,TRUE),0)
8	C1999772	=2020-MID(A8,2,4)	=IF(LEFT(A8,1) = "C","Citizen", IF(LEFT(A8,1) = "F", "Foreigner", "Permanent Resident"))	6200	2	=FLOOR(D8/E8,1)	=IF(OR(C8="Citizen",C8="Permanent Resident"),VLOOKUP(F8,$E$29:$G$31,3,TRUE),0)
9	C1947585	=2020-MID(A9,2,4)	=IF(LEFT(A9,1) = "C","Citizen", IF(LEFT(A9,1) = "F", "Foreigner", "Permanent Resident"))	1200	7	=FLOOR(D9/E9,1)	=IF(OR(C9="Citizen",C9="Permanent Resident"),VLOOKUP(F9,$E$29:$G$31,3,TRUE),0)
10	C1949500	=2020-MID(A10,2,4)	=IF(LEFT(A10,1) = "C","Citizen", IF(LEFT(A10,1) = "F", "Foreigner", "Permanent Resident"))	1600	5	=FLOOR(D10/E10,1)	=IF(OR(C10="Citizen",C10="Permanent Resident"),VLOOKUP(F10,$E$29:$G$31,3,TRUE),0)
11	C1978166	=2020-MID(A11,2,4)	=IF(LEFT(A11,1) = "C","Citizen", IF(LEFT(A11,1) = "F", "Foreigner", "Permanent Resident"))	5600	6	=FLOOR(D11/E11,1)	=IF(OR(C11="Citizen",C11="Permanent Resident"),VLOOKUP(F11,$E$29:$G$31,3,TRUE),0)
12	C2006054	=2020-MID(A12,2,4)	=IF(LEFT(A12,1) = "C","Citizen", IF(LEFT(A12,1) = "F", "Foreigner", "Permanent Resident"))	8000	4	=FLOOR(D12/E12,1)	=IF(OR(C12="Citizen",C12="Permanent Resident"),VLOOKUP(F12,$E$29:$G$31,3,TRUE),0)
13	F1988665	=2020-MID(A13,2,4)	=IF(LEFT(A13,1) = "C","Citizen", IF(LEFT(A13,1) = "F", "Foreigner", "Permanent Resident"))	10000	1	=FLOOR(D13/E13,1)	=IF(OR(C13="Citizen",C13="Permanent Resident"),VLOOKUP(F13,$E$29:$G$31,3,TRUE),0)
14	P1944818	=2020-MID(A14,2,4)	=IF(LEFT(A14,1) = "C","Citizen", IF(LEFT(A14,1) = "F", "Foreigner", "Permanent Resident"))	2300	7	=FLOOR(D14/E14,1)	=IF(OR(C14="Citizen",C14="Permanent Resident"),VLOOKUP(F14,$E$29:$G$31,3,TRUE),0)
15	C1980912	=2020-MID(A15,2,4)	=IF(LEFT(A15,1) = "C","Citizen", IF(LEFT(A15,1) = "F", "Foreigner", "Permanent Resident"))	1400	6	=FLOOR(D15/E15,1)	=IF(OR(C15="Citizen",C15="Permanent Resident"),VLOOKUP(F15,$E$29:$G$31,3,TRUE),0)
16	C2005380	=2020-MID(A16,2,4)	=IF(LEFT(A16,1) = "C","Citizen", IF(LEFT(A16,1) = "F", "Foreigner", "Permanent Resident"))	1250	2	=FLOOR(D16/E16,1)	=IF(OR(C16="Citizen",C16="Permanent Resident"),VLOOKUP(F16,$E$29:$G$31,3,TRUE),0)
17	C1977433	=2020-MID(A17,2,4)	=IF(LEFT(A17,1) = "C","Citizen", IF(LEFT(A17,1) = "F", "Foreigner", "Permanent Resident"))	900	4	=FLOOR(D17/E17,1)	=IF(OR(C17="Citizen",C17="Permanent Resident"),VLOOKUP(F17,$E$29:$G$31,3,TRUE),0)
18	F1959907	=2020-MID(A18,2,4)	=IF(LEFT(A18,1) = "C","Citizen", IF(LEFT(A18,1) = "F", "Foreigner", "Permanent Resident"))	1400	4	=FLOOR(D18/E18,1)	=IF(OR(C18="Citizen",C18="Permanent Resident"),VLOOKUP(F18,$E$29:$G$31,3,TRUE),0)
19	F1990499	=2020-MID(A19,2,4)	=IF(LEFT(A19,1) = "C","Citizen", IF(LEFT(A19,1) = "F", "Foreigner", "Permanent Resident"))	3600	2	=FLOOR(D19/E19,1)	=IF(OR(C19="Citizen",C19="Permanent Resident"),VLOOKUP(F19,$E$29:$G$31,3,TRUE),0)
20	P1971008	=2020-MID(A20,2,4)	=IF(LEFT(A20,1) = "C","Citizen", IF(LEFT(A20,1) = "F", "Foreigner", "Permanent Resident"))	4000	1	=FLOOR(D20/E20,1)	=IF(OR(C20="Citizen",C20="Permanent Resident"),VLOOKUP(F20,$E$29:$G$31,3,TRUE),0)
21	C1983766	=2020-MID(A21,2,4)	=IF(LEFT(A21,1) = "C","Citizen", IF(LEFT(A21,1) = "F", "Foreigner", "Permanent Resident"))	11500	1	=FLOOR(D21/E21,1)	=IF(OR(C21="Citizen",C21="Permanent Resident"),VLOOKUP(F21,$E$29:$G$31,3,TRUE),0)
22	F2006903	=2020-MID(A22,2,4)	=IF(LEFT(A22,1) = "C","Citizen", IF(LEFT(A22,1) = "F", "Foreigner", "Permanent Resident"))	3600	6	=FLOOR(D22/E22,1)	=IF(OR(C22="Citizen",C22="Permanent Resident"),VLOOKUP(F22,$E$29:$G$31,3,TRUE),0)
23	C1959344	=2020-MID(A23,2,4)	=IF(LEFT(A23,1) = "C","Citizen", IF(LEFT(A23,1) = "F", "Foreigner", "Permanent Resident"))	4800	3	=FLOOR(D23/E23,1)	=IF(OR(C23="Citizen",C23="Permanent Resident"),VLOOKUP(F23,$E$29:$G$31,3,TRUE),0)
24	C1978220	=2020-MID(A24,2,4)	=IF(LEFT(A24,1) = "C","Citizen", IF(LEFT(A24,1) = "F", "Foreigner", "Permanent Resident"))	2400	6	=FLOOR(D24/E24,1)	=IF(OR(C24="Citizen",C24="Permanent Resident"),VLOOKUP(F24,$E$29:$G$31,3,TRUE),0)

	A	B	C	D	E	F	G
27	**Monthly Per Capita Income**					**Per Capita Income Level**	
28	Median:	=MEDIAN(F5:F24)			**Monthly Per Capita Income**	**Description**	**Payout**
29	70th percentile:				0	Between $0 (inclusive) to $3,100 (non-	800
30					3100	Between $3,100 (inclusive) to $5,100	500
31					5100	$5,100 or more	100

Task 2:

```
num_inputs = int(input("Enter the number of sets of data to be
cateogorised: "))  # 2c[1]

for i in range(num_inputs):         # 2c[1]

    systolic = int(input("Enter your systolic pressure (mmHg): "))
    diastolic = int(input("Enter your diastolic pressure (mmHg): "))

    cat = None

    result = ["Normal BP", "High-normal BP", "Stage 1 Hypertension",
"Stage 2 Hypertension", "Isolated Systolic Hypertension"]     # 2b[1]

    print("Your BP is", systolic, "/", diastolic, "mmHg.")  # 2a[1]

    if systolic < 120 and diastolic < 80:
        cat = 0
    if (systolic >= 120 and systolic <= 139) or (diastolic >= 80 and
diastolic <= 89):
        cat = 1

    if (systolic >= 140 and systolic <= 159) or (diastolic >= 90 and
diastolic <= 99):
        cat = 2    # 2b[2]
    if systolic >= 160 or diastolic >= 100:
        cat = 3    # 2b[2]

    if systolic > 140 and diastolic < 90:
        cat = 4    # 2b[2]

    print("Diagnosis:", result[cat])
```

Task 3

```
all_hashes = ["#gogreen#recycling#upcycling",''
          "#coolgames#experiment#expertmode#awesomegames",''
          "#diycook#ironchef",''
          "#outdoor#fauna#flora#nature#scenery#sunset",''
          "#magic#howcanitbe"]

views = [230, 683, 388, 597, 127]

post_titles = ["My Upcycling Project", ''
               "Stanley's Awesome Games", ''
               "Chef @ Home", ''
               "An afternoon at Bukit Timah Reserve", ''
               "Card Tricks"]              #1 Missing ]

num_hashes = []

for i in range(5):   #2 Missing :
    num_hashes += [len(all_hashes[i].split("#")) - 1]
    #3 Typecast []

highest_hash = max(num_hashes)
index = None

for j in range(5):
    if num_hashes[j] == highest_hash:    #4 == not =
        index = j

#5 index instead of j
print(post_titles[index],"has",highest_hash,"hashtags which is
the highest among all posts.")

average = sum(num_hashes) / 5    #6     / instead of %

print("The average number of hashtags is", average)
#7 average instead of sum

highest_view = max(views)    #8 views instead of view
index = None

for k in range(5):  #9    5 instead of 4
    if views[k] == highest_view:
        index = k    #10    index = k instead of the other way

print(post_titles[index],"has",highest_view,"views which is the
highest among all posts.")
```

Task 4

11)

```
receipt = input("Enter your receipt number: ")
#[1] Initialize receipt and mobile

while len(receipt) != 9 or receipt[:4] != "DRAW" or
receipt[4:].isdigit()==False:
    receipt = input("Enter your receipt number: ")
#[2] Data validation of receipt

mobile = input("Enter your mobile phone number: ")

while len(mobile) != 8 or mobile.isdigit()==False:
    mobile = input("Enter your mobile phone number: ")
#[2] Data validation of mobile

prize_lst = ["a 65-inch television", "a $50 shopping voucher",
"a $10 shopping voucher", "a recylable bag"]
available = [1, 20, 50, 999999]
#[1] Initialize prize list and availability

import random
prize = random.randint(0,3)
#[1] Generate random number

while available[prize] == 0:
    prize = random.randint(0,3)
#[1] Choose random prize

available[prize] -= 1
#[1] Deduce prize balance

print("Congratulations! You have won", prize_lst[prize])
#[1] Display prize won
```

12) [4] for the 2 screenshots with correct data entered

Test Output 1:

```
Enter your receipt number: DR12345
Enter your receipt number: DRAW12345
Enter your mobile phone number: 12349876
Congratulations! You have won a <depending on random prize>
```

Test Output 2:

```
Enter your receipt number: DRAW23012
Enter your mobile phone number: 98123
Enter your mobile phone number: 98123456
Congratulations! You have won a <depending on random prize>
```

13)

```
if prize == 0:   #[1] 1st conditional statement
    print("Please double-check if your mobile number is",
mobile)
#[1] String concatenation and print

    reply = input("Enter 1 to confirm and 0 to re-input your
mobile number: ")
#[1] Ask user for input to confirm

    if reply == "1":   #[1] 2nd conditional statement
        print("You will be informed shortly through your mobile
number.")
#[1] Respond if confirm

    elif reply == "0":
        mobile = input("Please re-enter your mobile number: ")
        print("You will be informed shortly through your mobile
number.")
#[1] Request user to reenter
```