1 A junior college in Singapore has the following rules regarding reporting time for its students:

- On assembly days, students are to report to college before their first lesson if the first lesson is before 8.00am, otherwise they are to report to college by 8.25am for assembly.

- On non-assembly days, students are to report to college before their first lesson if the first lesson is before 9.30am, otherwise they are to report to college by 9.30am.

(a) Construct a decision table to show all possible combinations of conditions and outcomes. [4]

(b) Simplify your decision table to remove redundancies. [2]

(c) Write a program in pseudocode to check for the three conditions and return the reporting time for students according to the table in **(b)**. [3]

2 The recursive function `Power` has two parameters, `base` and `exponent`.

```
FUNCTION Power(base, exponent: INTEGER) RETURNS INTEGER
    IF exponent = 0
        THEN
            RETURN 1
        ELSE
            RETURN base * Power(base, exponent - 1)
    ENDIF
ENDFUNCTION
```

(a) State the features of a successful recursive function. [3]

(b) Create a trace tree diagram for the recursive function call `Power(2,2)`. [4]

(c) Describe the call stack operations that will take place for `Power(2,2)`. [4]

(d) State **one** example of an argument for exponent that would result in an infinite recursion. [1]

3 **(a)** Write an iterative function `sum_even_while(num1, num2)` that:

- takes in two non-negative integers
- computes the sum of even numbers from `num1` to `num2` using while loops
- returns the result. [5]

Call your function using the following statement:
`print(sum_even_while(0, 10))` [1]

(b) Write a recursive function `sum_even_r(num1, num2)` that does the same computation as your function in **(a)**. [5]

Verify your answer in **(b)** using the following statement:
`print(sum_even_r(0,10) == sum_even_while(0,10))` [1]

(c) Write a recursive function `sum_all(num1, num2, odd_even)` that:

- takes in two non-negative integers `num1` and `num2` and a third integer `odd_even`, which takes on an integer value of either 0 or 1, representing odd or even respectively
- computes the sum of all even or odd numbers from `num1` to `num2` inclusive
- returns the result. [6]

Call your function using the following statements:

```
print(sum_all(0, 10, 0))
print(sum_all(0, 10, 1))
```

[2]

4 **(a)** Write a function `read_from_file(filename)` that takes the name of any text file as input argument, reads the data from the file, and returns the data as a list of integers. [3]

The file `random_integers.txt` contains 1000 randomly generated integers between 1 to 10000, inclusive of the two bound values. Each line contains 1 integer.

(b) Read the integers stored in `random_integers.txt` using your function in **(a)** to a list `random_values`.

(c) Sort the integers in random_values in descending order using

(i) bubble sort
(ii) insertion sort
(iii) merge sort