Task 1a explanation

- First, an input file was taken which is then read.
- Using the readlines(), we made a list and stored in variable.
- We ran a loop over it, checking skipping the first line of the text file, checking whether the input after it is odd or even.

- Each output is appended to a list.

- We write on the output text file by running a loop through the list.

Task 1b explanation

- An input file taken with inputs written on it already provided, it is then read.

- Readlines() method used to make a list.

- We run a loop through it, skipping the first line, second line is read, we convert the numbers from string to integer, through that correct operation taken to evaluate the value of the two operands which is then appended to a list.

— We write on the output file by
looping over the above list mentioned.

Task 2

— Bubble sort algorithm used to sort
the numbers in ascending order.

— Flag used in between the loops and set
it as False.

— Then, the two numbers are compared,
and necessary swapping takes place,
setting the flag to True.

— When the array is sorted, flag will be
always be false.

— If flag value remains false, then
the loop breaks.

— The output is then write on the
output file

Task 3

- Input file taken
- Stable ^Selection sort algorithm used to sort the marks in descending order.

- When the marks are placed at it's correct position, id is also changed to the changed position.

- A loop is run over the sorted marks list, when the marks are the same, their ids are compared, the id with the same value are placed first then the id with the low value.

- we write on the output file

Task 4

- We take an input file which is read.
- Three list initialized to store the each line, train names & time of departure

- Loop is run through and each line stored which is then append to a variable, train name & time taken while we split the line variable.

- Bubble algorithm used to compare between train names in as to maintain lexicographic order ...., if the train name remains the same, then time is compared, latest time prioritized. However, time can also be tied which can be arranged in a way that the time which comes first in the input file will be prioritized.

- Then, we write on the output file.