# WEEK01

**I will skip the description of opening file and writing file in txt file.**

## TASK 1A:

*Read the 1st line to know number of items
*Read all the lines
*for loop converting all the items in integer and removed “\n” by using rstrip()

**odd_even() function to return the result
* if it’s modded by 2 then it will return it’s even else it will return it’s odd

*open variable for adding all the results
*for loop for adding all the number by calling the odd_even() again and again

## TASK 1B:

*Read the 1st line for the size
*Read all the lines
*removing ‘calculate ’ and ‘\n’ by using rstrip() and lstrip() by looping through all the items

*calculator() function to return the result
* splitting the items [ex: “13 + 12” → “13”, “+”, “12”]
*if, elif, else condition for +, -, *, / operands
*converting the strings to integer and initiating the operation
*returning the result with information and final product with proper strings

*Declaring an empty string variable
*adding all the results by looping and calling the calculator() function to the empty string

## TASK 02:

*Reading 1st line for the size
*Reading rest of the lines which are the numbers
*splitting and converting the numbers from string to integer using for loop

*BubbleSort() function which returns the sorted array
*first for loop for each element. Range till size -1
*flag == False for keeping track of swapping
*second for loop to compare the other elements with 1st loop’s index. Range would be i+1 to size

*now an if condition to check the other element is bigger or not
*swapping if they are big and flag would be true
*checking if items are swapped or not by using the flag. [It's for better optimization which saves time for sorted arrays.]

*Declaring an empty string variable to add the results by using for loop

**TASK 03:**

*Reading the 1st line to side out the size
*List of IDs
*List of Marks
*making all the data in ID and Marks integer using for loop

*Creating sorted_marks() to sort both data
*I am using selection sort to sort the data
*Using 1st for loop to hold the index slot. Range till size -1
*A variable to compare index
*Second loop for going through the rest of the index. Range till i+1 to size
*1st condition if the current index and the check index matches the condition. It will update the pointing variable. Which will sort the data in decreasing order.
*The 2nd condition will work if the 1st data is equal. It will create the 2nd data in ascending order.
*Next with the updated index it will swap the data by using a temporary variable.
*It will return both data in a tuple

*By unpacking It will be added in an empty string variable using a for loop.

**TASK 04:**

*Reading the 1st line for the size
*declaring three variables with empty lists for three types of data
*By using the for loop it will read each line and split the data with using spaces
*After splitting the data each 3 data will be divided by indexing the strings and will get appended to the lists

*parse_time() function will take the data from the time list and make it comparable by splitting and adding both hour and minute data by converting it to integer

*sorted_train() function is also made in selection sort process
*Using 1st for loop to hold the index slot. Range till size -1
*A variable to compare index
*Second loop for going through the rest of the index. Range till i+1 to size

*1st condition if the current index and the check index matches the condition(lexicography). It will update the pointing variable. Which will sort the data in ascending order.
*The 2nd condition will work if the 1st data is equal. It will create the 2nd data in descending order.(comparing time using parse_time())
*Next with the updated index it will swap the data by using a temporary variable.
*It will return three data in a tuple

**By unpacking It will be added in an empty string variable using a for loop.