MOD 11 Check Digit

A check digit is a number that is used to validate a series of numbers whose accuracy you want to insure. Frequently the last digit of a number string such as identification number is a check digit. Lets say the identification number starts out at 6 digits. A calculation is done using the six digits and a seventh digit is produced as a result of the calculation. This number is the check digit. There are many calculations that can be used - this example illustrates the logic of the MOD11 check digit.

Steps to calculate the MOD11 check digit for a number such as an id #:

- Assign weights to each digit of the id #. The weights in MOD11 are from 2 through a maximum of 10 beginning with the low order position in the field.
- Each digit in the id # is multiplied by its weight
- The results of the multiplication are added together
- This product is divided by the modulus number 11
- The remainder is subtracted from the modulus number 11 giving the check digit

Example: find the check digit for the number **036532**

0	3	6	5	3	2
x7	x6	x5	x4	x3	x2
0	18	30	20	9	4

```
0 + 18 + 30 + 20 + 9 + 4 = 81

81/11 = 7 remainder 4

11 - 4 = 7
```

7 is therefore the check digit.

PROBLEMS: If the remainder from the division is 0 or 1, then the subtraction will yield a two digit number of either 10 or 11. This won't work, so if the check digit is 10, then X is frequently used as the check digit and if the check digit is 11 then 0 is used as the check digit. If X is used, then the field for the check digit has to be defined as character (PIC X) or there will be a numeric problem.

Steps to verify if the check digit is included as part of the number:

- The entire number is multiplied by the same weights that were used to calculate and the check digit itself is multiplied by 1.
- The results of the multiplication are added together.
- The sum is divided by 11 and if the remainder is 0, the number is correct.

PROBLEM: Note that if the check digit is X then 10 is used in the multiplication. Code for this occurrence must be included.

Example of verifying the number **0365327** where 7 is the calculated MOD11 check digit:

0	3	6	5	3	2	7
x7	x6	x5	x4	x3	x2	x1
0	18	30	20	9	4	7

```
0 + 18 + 30 + 20 + 9 + 4 + 7 = 88

88/11 is 8 remainder 0
```

Since the remainder from this calculation is 0, the check digit 7 is valid.

Question 3 (24 marks) suggested time 1 hours 10 mins

Filename submitted S31Cxxxxxxxx.py where xxxxxxxx is your student number

BU are developing the application for processing student information. In this program, you need to do the following:

1) *Provide comments* for your program written (4 mark)
2) The function of *producing the Hash Total* (6 mark)
3) The function to *producing the check digit* (6 marks)
4) The procedure for *accepting input of data* (4 mark)
5) The procedure to *printout the result* (4 marks)

***Anything you want to inform the marker about your program, **please** written down in the comments of your program*

After capture the number **NoOfdata** (maximum 5); then capture the **AppNo**
your student ID can be a constant defined in the program **OR** variable entered

Suggested output of the program

Operator: your student ID (CDK) : Check Digit Key (HTK) : Hash Total Key
Hash total of the data: Hashtotal Number of data entered : NoOfdata
then a list of all AppNo; and NewAppNo

The functions can be a separate module OR inside the same program

Program Requirement

Encrypt the data enter and add the operator identity to serve as responsibility checking, below is the detail calculation required:

All the **AppNo** is 4 digits

1. Assume the *maximum* **AppNo** inputted are **5 only**; **NoOfdata** is the number inputted by user to determine the number of data entered.
2. For the **AppNo**, using the below description** to calculate the check digit to form the **NewAppNo**
3. Using the 3rd to the 6th digit in your student ID as **Check Digit Key**, For example, assume **you student ID** is 20201778 (20**2017**78); so **2017** is **Check Digit Key**
4. Using the 5th to the 8th digit in your student ID as Hash Total Key, For example, assume you student ID is 20201778 (20201778); so 1778 is Hash Total Key

Logic to form the check digit (range from 0 to 6)

** assume one of the **AppNo** entered is 2345, multiplier is your Check Digit Key (in the above case is 2017) ---- one to one corresponding

2		3		4		5
*	+	*	+	*	+	*
2		0		1		7

+ Check Digit Key (2017 in this example) + check digit is divisible by 7

so 43 + 2017 + check digit** is divisible by **7**, the **check digit** should be **5 *(2065 is divisible by 7)*

So the **NewAppNo** is 23455 ← 5 is the check digit add to the suffix of this AppNo 2345

Logic for the HashTotal

@@ Suppose there are only 2 data enter, the **AppNo** are 1234 and 3456, then 1234 + 3456 = **TotalAppNo** is 4690; your Hash Total Key is 1778 and then

HashTotal is = (Hash Total Key) 1778 + 4690 = 6468

Remark: You have *NewAppNo for each AppNo entered* but you *only have 1 Hash Total* for the batch

*[**Bonus mark** : You can provide flexibility in your program …]*