

Part 1. [10pts]

Given two integers X and Y, you will calculate the first integer (f_i) which can be divided by the given third number, Z. If there is not, you will return -1 as an exit code to show the message "There is not any integer between X and Y can be divided by Z". Otherwise, you will also calculate the Nth number, where it is given by the user to specify the dividable Nth number. (If the N is 2, then you will calculate the 3rd number). Be careful that you should handle the range of Nth number, again you may use -999 as out of bounds error code. For this error code, you will print, "No possible to find Nth divisible between X and Y divided by Z".

You will ask for the user input in the main function and pass the parameters to the find_divisible and find_nth_divisible methods. These functions will return to the main function, and the main function handles the outputs that will be printed on the command prompt. Please investigate all the cases carefully, some of the examples are given below.

Example 1:

Enter the first integer: 10

Enter the second integer: 20

Enter the divisor: 3

Output> The first integer between 10 and 20 divided by 3 is 12

Enter the number how many next: 2

Output> The 3rd integer between 10 and 20 divided by 3 is 18

Example 2:

Enter the first integer: 20

Enter the second integer: 10

Enter the divisor: 3

Output> There is not any integer between 20 and 10 can be divided by 3

Example 3:

Enter the first integer: 18

Enter the second integer: 20

Enter the divisor: 11

Output> There is not any integer between 18 and 20 can be divided by 11

Function prototype is `int find_divisible(int x, int y, int z)`

Function prototype is `int find_nth_divisible(int n, int f_l, int z)`

Part 2. [60pts]**Create a new customer by identity number**

Note: For all the functions below, you may use 1 and 0 for correct and incorrect return codes, respectively.

Suppose that you are a programmer behind ATMs, and a wily person tries to create a new account by using random numbers as an identity number and wants to transfer his cash by withdrawing from his blocked account. However, ATMs must have a format validation process for given identity numbers.

For the identity number entered by the customer,

- **You must write a c function that validates the format of the identity number by following these rules below.**

Function prototype is `int validate_identity_number(char identity_number [])`.

1 – TR Identity Numbers must be 11 characters.

2 – Each digit must be a digit.

3 – The first digit cannot be 0 (zero)

4 – When we subtract the sum of the digits 2, 4, 6, 8 from 7 times the sum of the digits 1, 3, 5, 7, 9, the remainder of the division by 10 (MOD10) should give the number in the 10th digit.

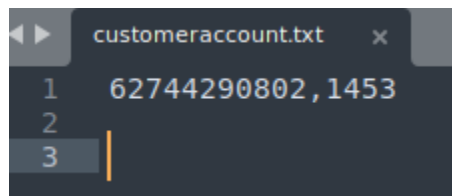
5 – The remainder of the division by 10 of the result obtained from the sum of the first 10 digits (MOD10) should give the number in the 11th digit.

the algorithm pseudo code, <https://dinamiknetwork.com/tc-kimlik-no-dogrulama-algoritmasi/>

Identity number generator, <https://www.simlict.com/>

- For the validated identity numbers, the password (4 digits) given by the customer as an additional customer authorization information must be saved into the file named "customeraccount.txt". Finally, the file contents should be shown like in the figure below (identity_number,password). The identity number shown in the file generated by the TC identity number generator.
- We assume that your file contains only one customer for this homework.

Function prototype is `int create_customer(char identity_number [], int password)`



Part 3. [30pts]

In this part you will use the created customeraccounts.txt file already in part 2 as your database to authorize the login on the ATMs.

Function prototype is `int check_login(char identity_number [], int password)`

After a successful login, suppose that ATM has 10, 20 and 50 liras to cash withdraw. Write an additional function that calculates the withdrawable amount for a requested cash amount. then provide the withdrawable amount for the customer. We assume that the available balance of the customer is enough to withdraw.

Function prototype is `int withdrawable_amount(float cash_amount)`

Example 1:

Enter your identity number: 10000000000

Enter your password: 1453

Output> "Invalid identity number or password"

Example 2:

Enter your identity number: 62744290802

Enter your password: 1234

Output> "Invalid identity number or password"

Example 3:

Enter your identity number: 62744290802

Enter your password: 1453

Output> "Login Successful"

Enter your withdraw amount: 180.75

Output> Withdrawable amount is: 180

Example 4:

Enter your identity number: 62744290802

Enter your password: 1453

Output> "Login Successful"

Enter your withdraw amount: 185

Output> Withdrawable amount is: 180