**APSIT SKILLS INTERNSHIP – PROJECT REPORT**

**SE COMPUTER**

Project Batch : B2

Team Members Names -

Team Leader :- RIDDHI NARKAR

Member 1 :- ADITYA YADAV

Member 2 :- HET PATEL

Technology Selected :PYTHON

Project Topic Name : SOLVING MATH USING PYTHON.

Problem Statement:

Solving math using Python set 2.

1. Unit Converter.
2. Calculate a simple Quadratic Equation.
3. Find the largest element amongst the list.
4. Create a multiplication table, asking about which multiplication table user wants and till which number user wants to create the multiplication table.
5. Python program to convert Decimal to Binary, Octal and Hexadecimal and vice versa.
6. Find Sum, Difference and Multiplication of list of numbers.
7. Find whether a number is divisible by another number or not.
8. To check whether the number is an Armstrong number or not.

Detailed Workflow:

* Functions/modules used for the program
* Modules -
* Functions -
* Flow of you code
* To achieve the desired output, we will be following the approach of Menu-Driven methods where we will display a menu on the initial page with multiple choices as:

1. Unit Converter.
2. Quadratic equation solver.
3. Finding the largest number.
4. Multiplication table calculator.
5. Number system converter.
6. Basic math operations.
7. Divisibility test.
8. Armstrong’s number.
9. Exit.

* If incase the user enters other number, then an appropriate error message will be prompted.

All the functions will be displayed on main menu page, here the input chose, will call for respective command to run and ultimately the user will be directed towards his desired option.

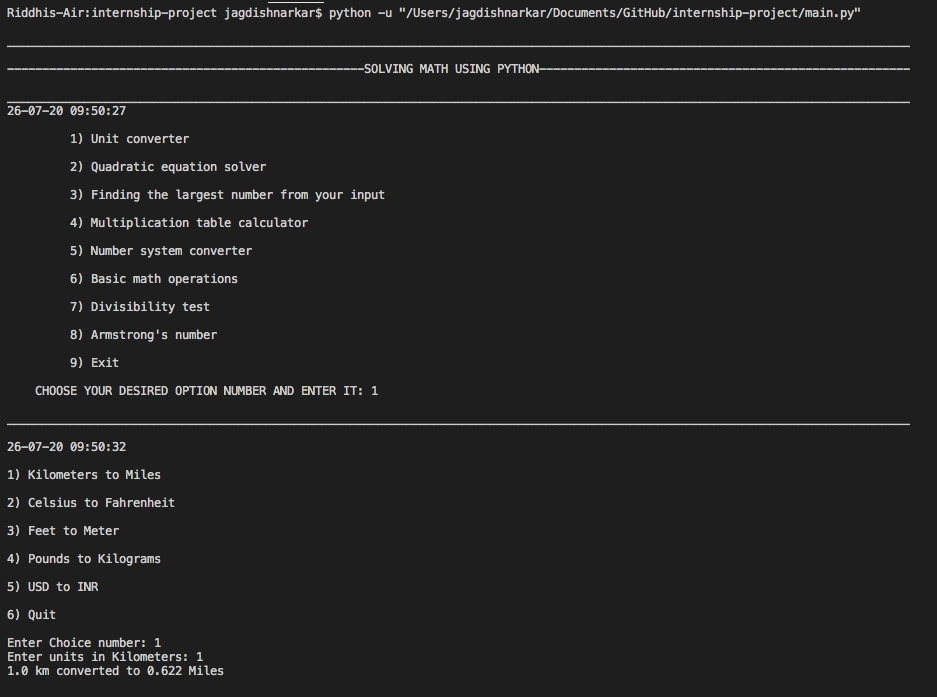
Here we have used the function def\_name() to create sub-functions and this sub-functions will be embedded in the main() function, by simply adding the def\_name(). Main menu page will run on the principle of if-elif conditional statements.

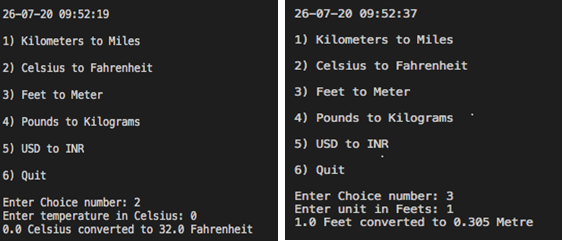
1. Unit Converter.
2. Here user is given with 5 sub types of unit conversions. The program will prompt, for which unit conversion user wants to do.
3. After selecting the desired option, user will be prompted to give inputs of that particular unit. The units will be converted to floating input.
4. Then the program will print out the result of conversion.
5. If user wants to select different unit converter then the if-elif loop will iterate and user is expected to enter the option which he wants.
6. If user chose other option then specified, then the if-elif statement will eventually turn false and error message will be displayed.
7. If user wants to go back to main menu then he can enter ‘6’ which will redirect the user.
8. Quadratic Equation Solver.
9. The inputs expected in this set of program will be the coefficients a, b and c with a space to differentiate the coefficient.
10. Firstly, we will be importing the math function (import math) and then initialize the value of discriminant (D) with the formula.
11. Then with the help of if-elif statement, we will be creating the possible outcomes condition.
12. Depending upon the value of Discriminant, roots will be printed.
13. If the value of Discriminant is unreal then an error message will be prompted.
14. After the execution of code user will be redirected to the main menu page.
15. Finding the largest number.
16. In this problem statement, we will be using the concept of list[] and will initialize the value of variable(n) to be zero.
17. Then it will prompt user to, ‘enter number of elements’ and then it will store it in that initialized variable(n).
18. After that it will prompt user to add the numbers which he wants to test.
19. After this, for loop will iterate till the range of (0,n) and to check the values entered are true we have also used while loop.
20. The new concept of Try and Except is also being used, if incase there is an error then too program won’t be aborted.
21. If the input doesn’t match the criteria than an error message will be shown.
22. To print the maximum among the given number list we have used the function max(list).
23. After the execution of code user will be redirected to the main menu page.
24. Multiplication table calculator.
25. In this program, user is expected to enter multiplication of any number he wants and will also ask about till which number he wants to get answer.
26. Using the for loop we will be defining the range of the program.
27. Once the inputs are taken, it will printout the result in a multiplication table format.
28. Number system converter.
29. User is expected to select conversion type i.e., from decimal to binary, octal, hexadecimal or binary, octal, hexadecimal to decimal.
30. After selecting one of the option program takes input from the user.
31. Using if-else ladder desired conversion is outputted based on users selection.
32. For decimal to binary, octal and hexadecimal we have a while loop for storing the remainder in a list.
33. When the while is terminated prefix is appended in the list and using inbuilt reverse function, list is reversed and is converted to string for loop.
34. Program prints the converted number.
35. For binary, octal and hexadecimal to decimal we have used inbuilt int() function for the conversion.
36. Based on the users selection decimal number is converted to the base 2,8 and 16 respectively using int( ).
37. Basic math operations.
38. Here user is expected to choose between 4 operations i.e., addition, subtraction, multiplication and subtraction of 2 lists.
39. After selection of preferred option, inputs will be appended in the list.
40. Using if-else statement user can choose among the following math operations he desires.
41. For addition, in built function sum(list) is used.
42. For subtraction, we had used a logical operator -=variable\_name.
43. For multiplication, we had used \* operator and had already initialized variable to 1.
44. For subtraction of 2 lists, we prompt the user to enter another list of the same length as before.
45. We input that list using try and except style, and a string is thrown if the input is not numerical to change the input.
46. For the output we take the difference of corresponding elements of the two lists in proper order and put this in the result list using list comprehension and output the same.
47. An error message be prompted if the condition isn’t satisfying.
48. Divisibility Test.
49. User is expected to enter a number and divisor.
50. Using if-else statement and modulus (%) operator program checks whether the given number is divisible by the divisor.
51. Armstrong’s Number.
52. Program prompts user to enter a number.
53. Using list comprehension digits of number is converted into a list.
54. Now the sum of cubes of all the digits is calculated using a for loop.
55. Using if-else statement, if the sum of cubes of the digits is equal to the original number then program prints 'the given number is an Armstrong number’ else not an Armstrong number.

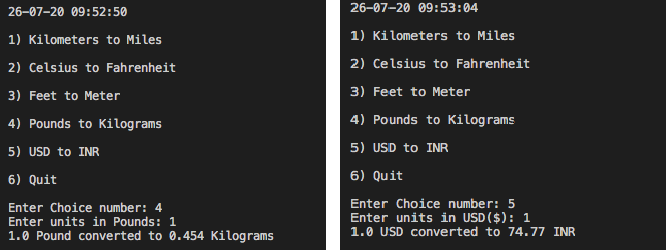
GitHub link of project:

Output Screenshots:

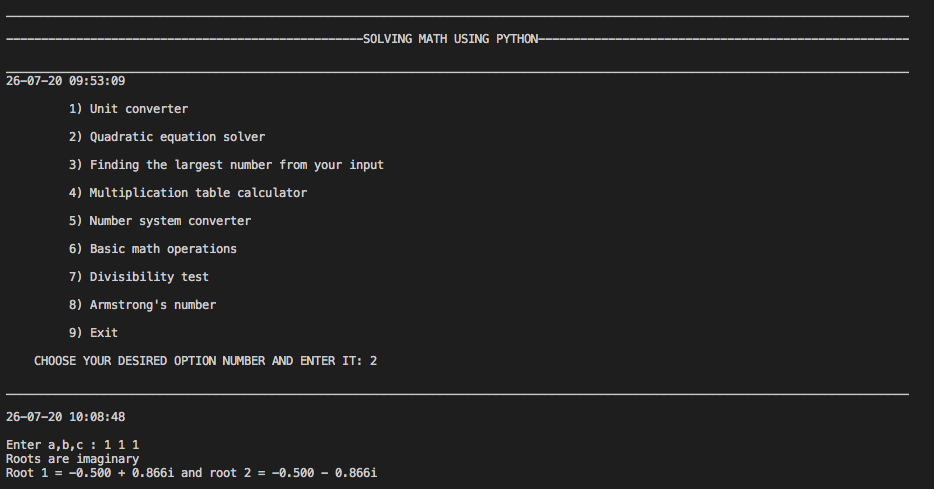
1. Unit Converter.

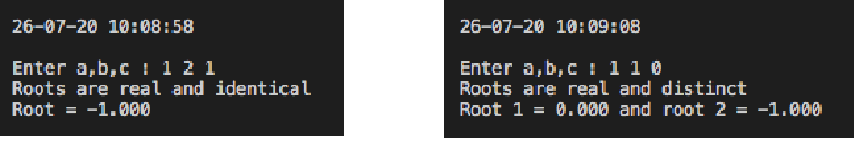


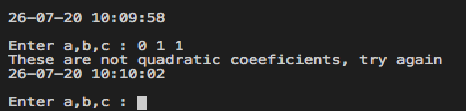




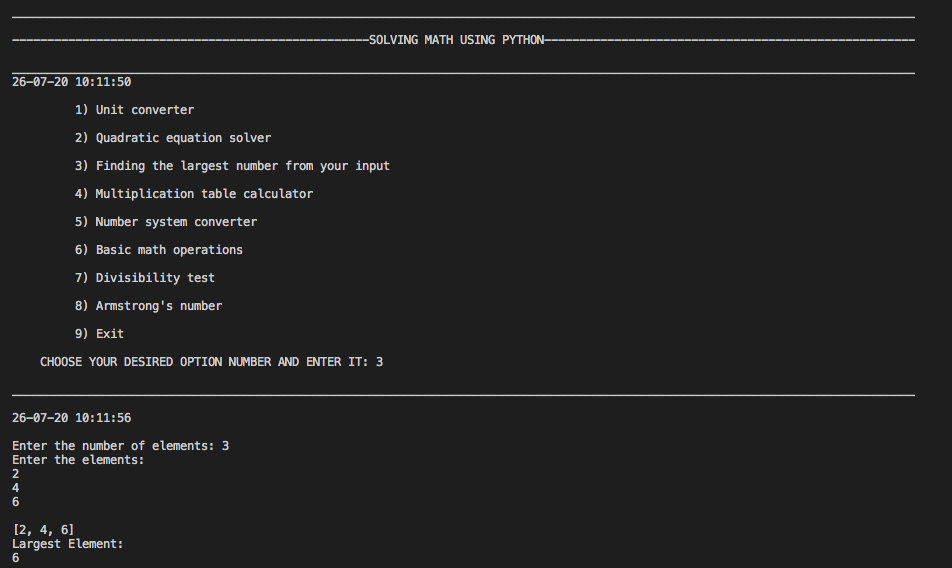
1. Quadratic Equation Solver.

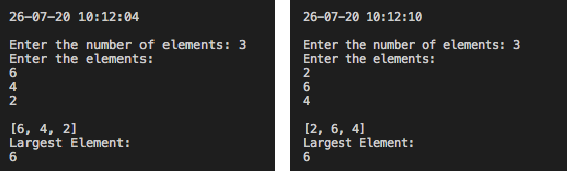




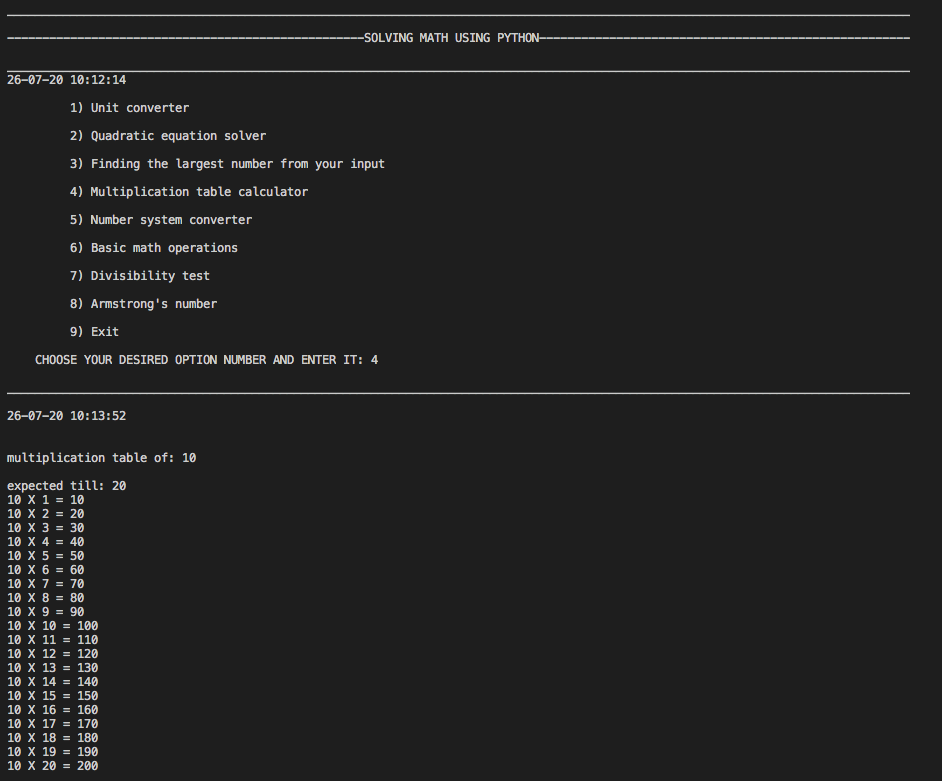


1. Finding the Largest Element.

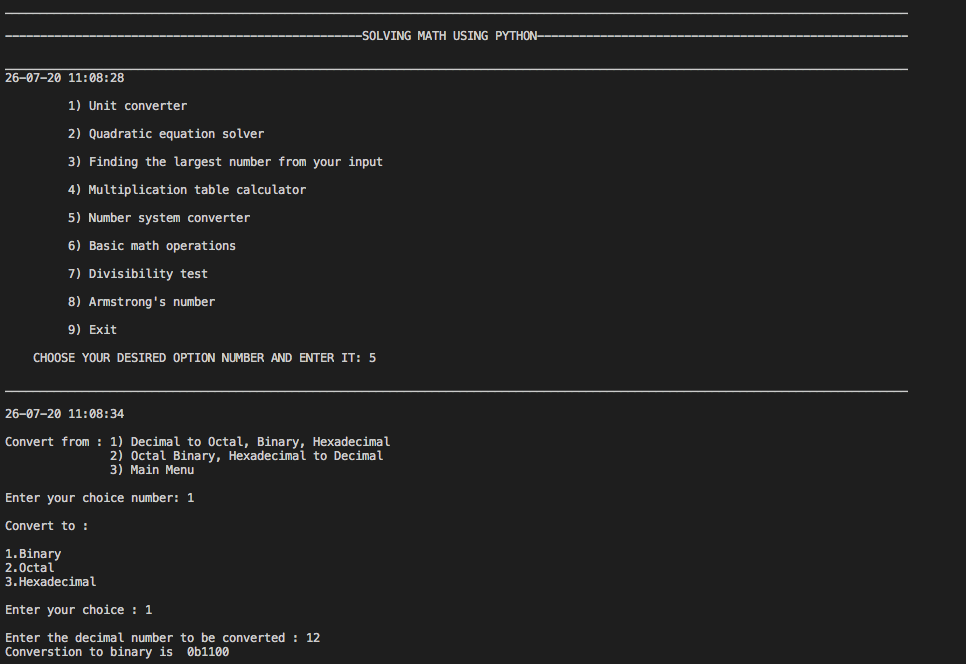


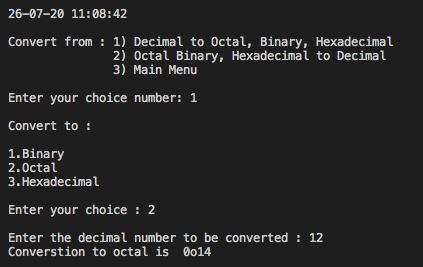


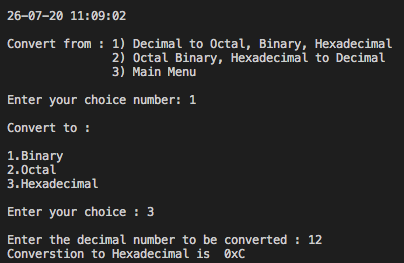
1. Multiplication table calculator.

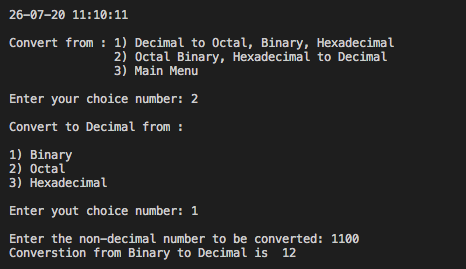


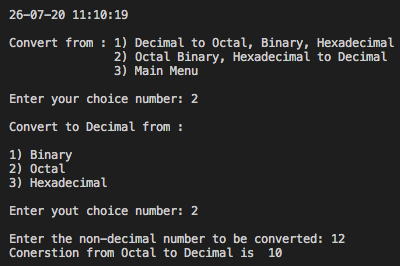
1. Number system converter.

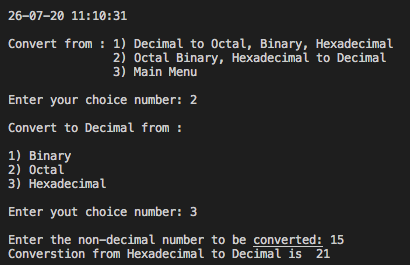




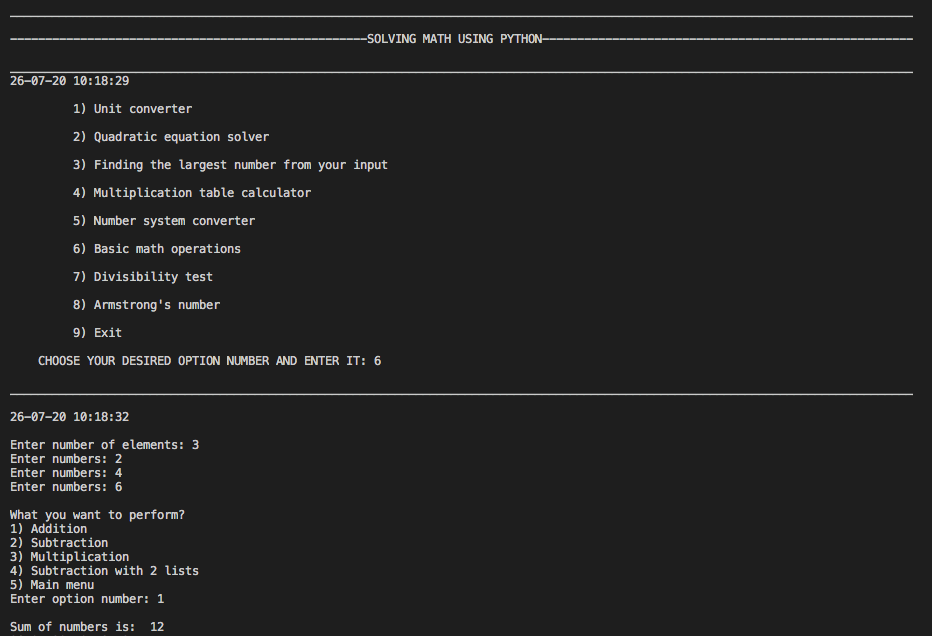


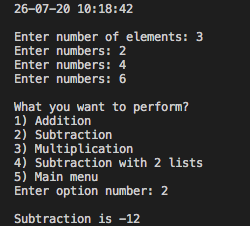
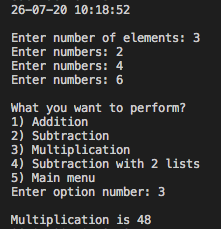


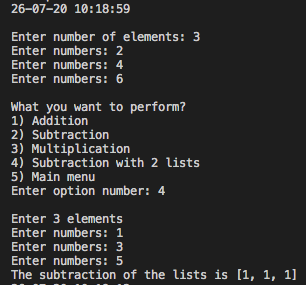


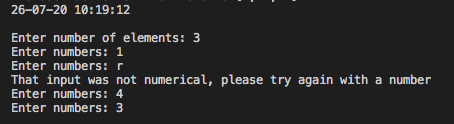


1. Basic Math Operations.

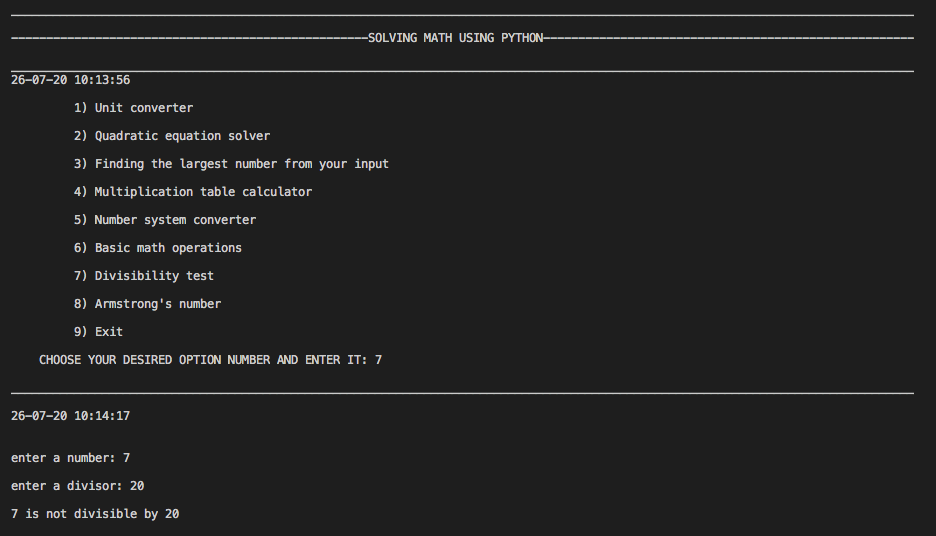


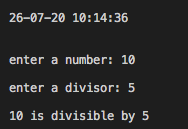
 



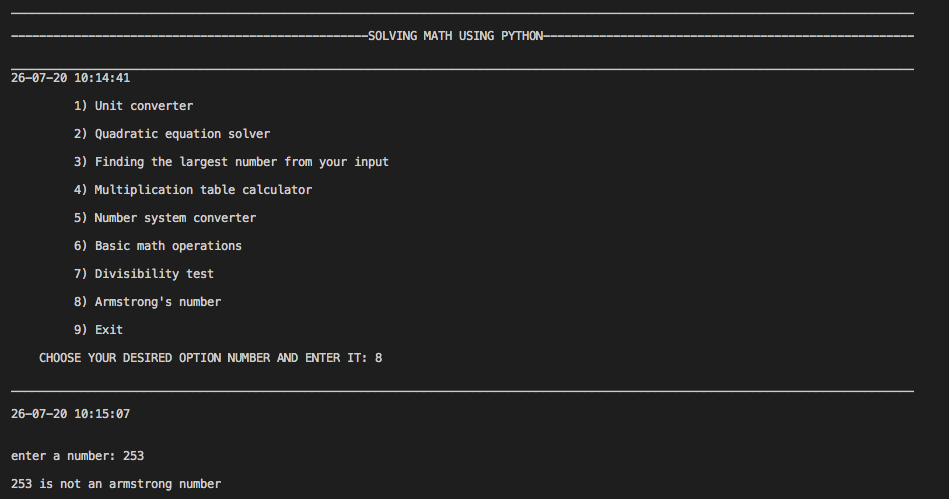


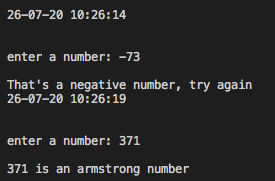
1. Divisibility test.





1. Armstrong’s Number.





Acknowledgment :