**Department of Computer Engineering**

**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 SET 2

**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, Multiplication on a list and difference of 2 lists.
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 – datetime.py, math.py, max(), int()
* Functions – main(), time(), unit\_convert(), quadratic(), largest(), mult\_tab(), num\_convert(), basic\_math(), divisible(), and Armstrong().
* **Flow of your code**
* To achieve the desired output, we will be following a Menu-Driven approach 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 chosen input 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.

With a view to make this code more user-friendly, we have tried to develop this code with exception handling. For this we have implemented the concept of try and except.

1. **Unit Converter:**
2. Here, the user is provided with 5 types of unit conversions namely (1) Kilometres to Miles, (2) Celsius to Fahrenheit, (3) Feets to Metres, (4) Pounds to Kilograms, and (5) USD to INR. 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 can re-enter the option which he desires.
6. If user enters any character other than the option number, 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 between each 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 possible outcomes condition.
12. Depending upon the value of Discriminant (D), roots will be printed.

(1) D>0: real, distinct roots

(2) D<0: imaginary, conjugate roots

(3) D=0: real, identical roots

1. If the value of the coefficient ‘a’ is 0, then an error message will prompt the user to re-enter the input.
2. After the execution of code user will be redirected to the main menu page.
3. **Finding the largest number:**
4. In this problem statement, we will be using the concept of list[] and will initialize the value of one variable(n) to be zero.
5. Then it will prompt user to, ‘enter number of elements’ and then it will store it in that initialized variable(n).
6. After that it will prompt user to add the numbers which he wants to test.
7. After this, for loop will iterate till the range of (0,n) and to check the values entered are numerical we have also used while loop.
8. The new concept of Try and Except is also being used, if incase the input isn’t numerical, then too program won’t be aborted, rather an error message will inform the user and will ask of a new proper input. Also here, the previous input(s), if any, won’t be lost.
9. To print the maximum among the given number list we have used the function max(list).
10. After the execution of code user will be redirected to the main menu page.
11. **Multiplication table calculator:**
12. In this program, user is expected to enter multiplication of any desired number and also till which integer he wants the multiplication.
13. Using the for loop we will be defining the range of the program.
14. Once the inputs are taken, it will printout the result in a proper multiplication table format. To achieve this, we used the format function with print.
15. **Number system converter:**
16. User is expected to select conversion type i.e., either from decimal to (binary, octal, hexadecimal) or (binary, octal, hexadecimal) to decimal.
17. After selecting one of the options, program takes input from the user.
18. Using if-else ladder desired conversion is displayed as output based on user’s selection.
19. For decimal to (binary, octal and hexadecimal) we have a while loop for storing the remainder in a list.
20. When the while is terminated, using inbuilt reverse function, list is reversed and is converted to string via a for loop.
21. Program prints the converted number.
22. For (binary, octal and hexadecimal) to decimal we have used inbuilt int() function for the conversion.
23. Based on the users selection decimal number is converted according to the bases: 2,8 and 16 respectively using int( ).
24. **Basic math operations:**
25. Here user is expected to choose between 4 operations i.e., addition, difference, multiplication and difference of 2 lists. Error is thrown if the user enters any other character rather than the option numbers, and the user will be prompted to re-enter the same.
26. After selection of preferred option, inputs will be appended in the list.
27. Using if-else statement user can choose among the following math operations he desires.
28. For addition, in built function sum(list) is used.
29. For subtraction, we had initialized a variable to 0 and subtracted all the elements from it using a shorthand operator for subtraction in a for loop.
30. For multiplication, we had used \* operator and had

initialized one variable to 1, which is then multiplied with each element of the list.

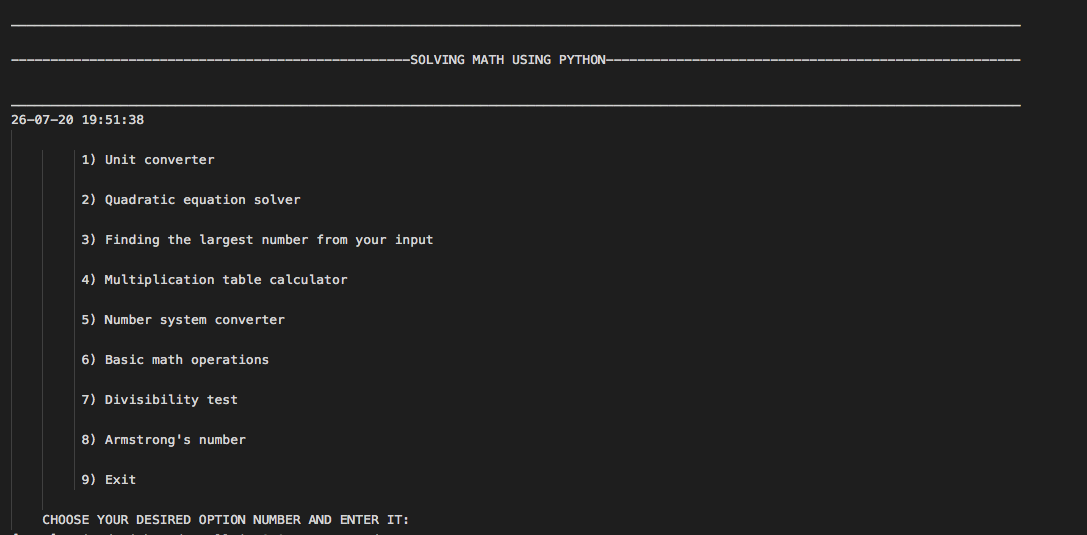
1. For subtraction of 2 lists, we prompt the user to enter another list of the same length as before.
2. We input all lists using try and except style, and a string is thrown to change the input if it is not numerical.
3. For the output we take the difference of corresponding elements of the two lists in proper order and put this in the result list using list comprehension and output the same.
4. **Divisibility Test:**
5. User is expected to enter a number and divisor.
6. Using if-else statement and modulus (%) operator, the program checks whether or not the given number is divisible by the divisor.
7. **Armstrong’s Number:**
8. Program prompts user to enter a number.
9. Using list comprehension, individual digits of number are converted into a list.
10. Now the sum of cubes of all the digits is calculated using a for loop.
11. 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.

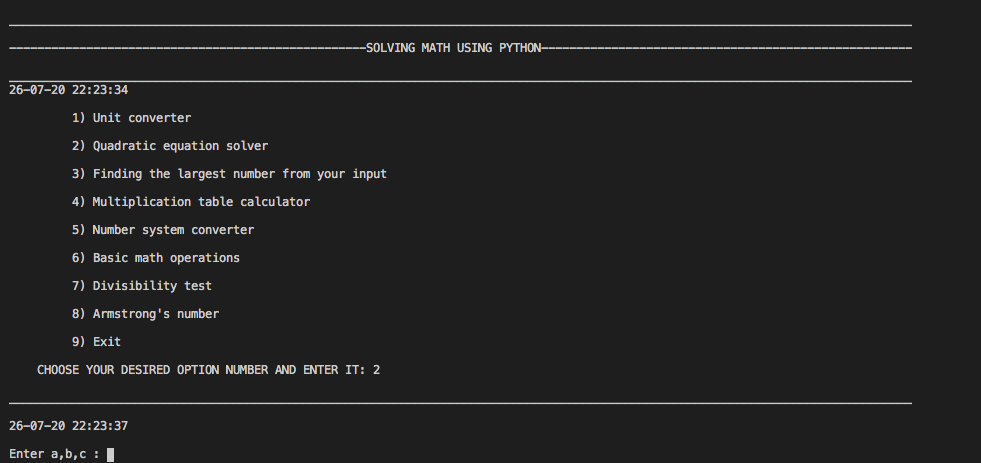
**GitHub link of project:**

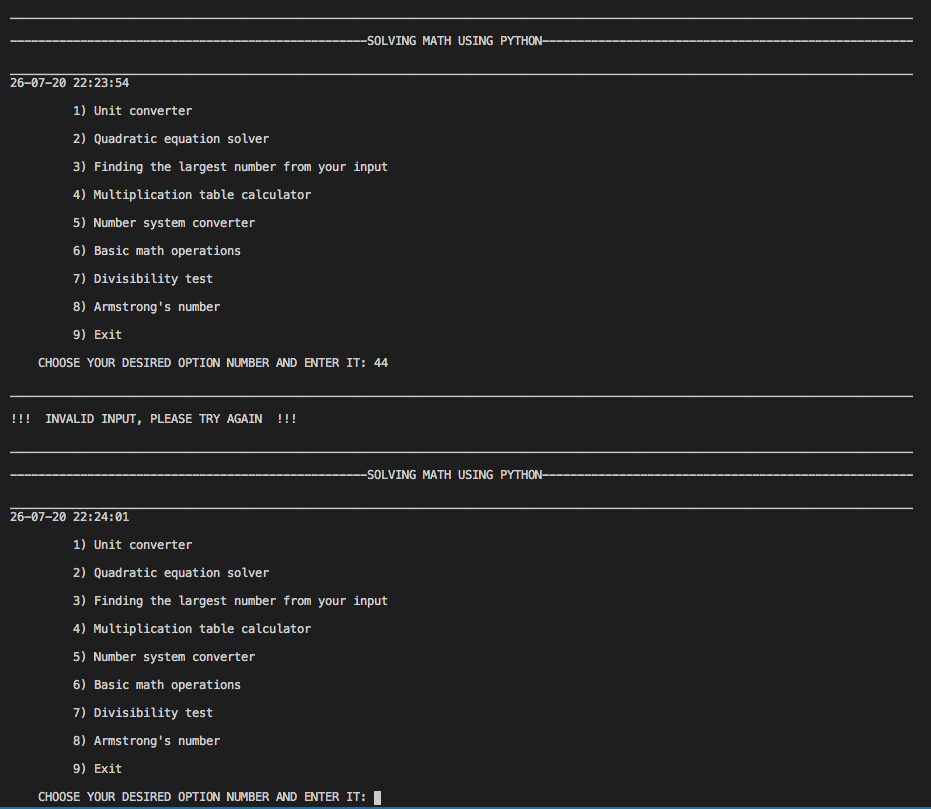
[GitHub repository: internship-project](https://github.com/grey-hat2003/internship-project)

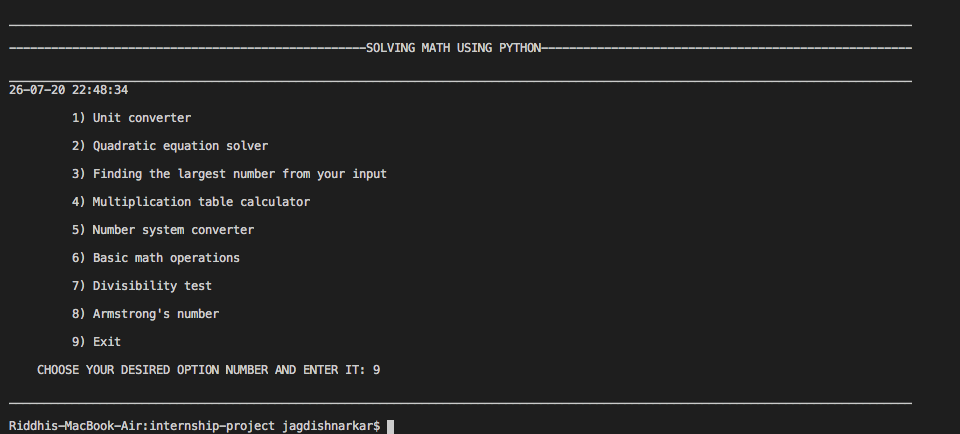
**Output Screenshots:**

1. Main menu:

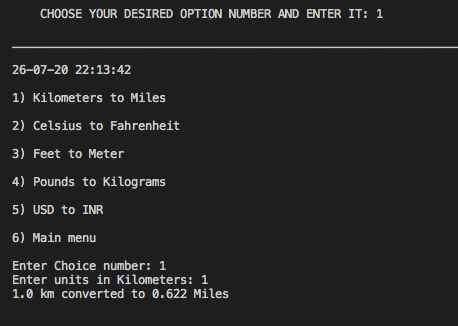
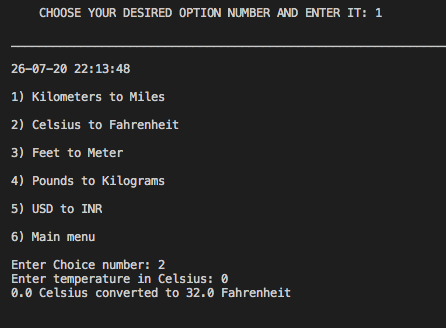


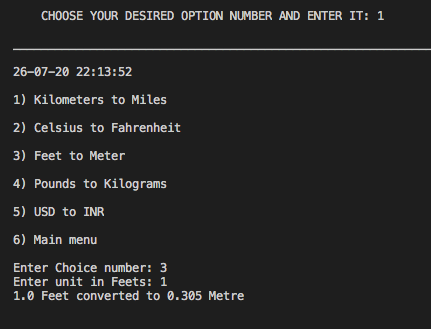
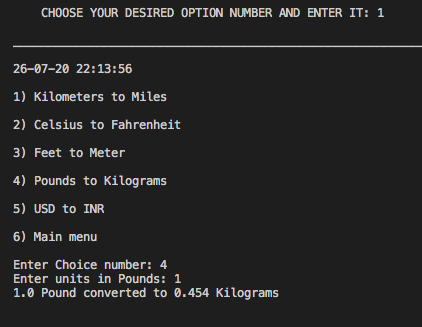


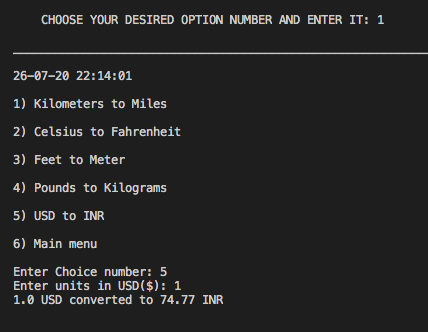


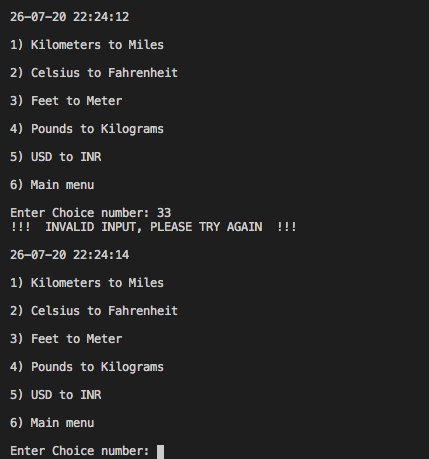


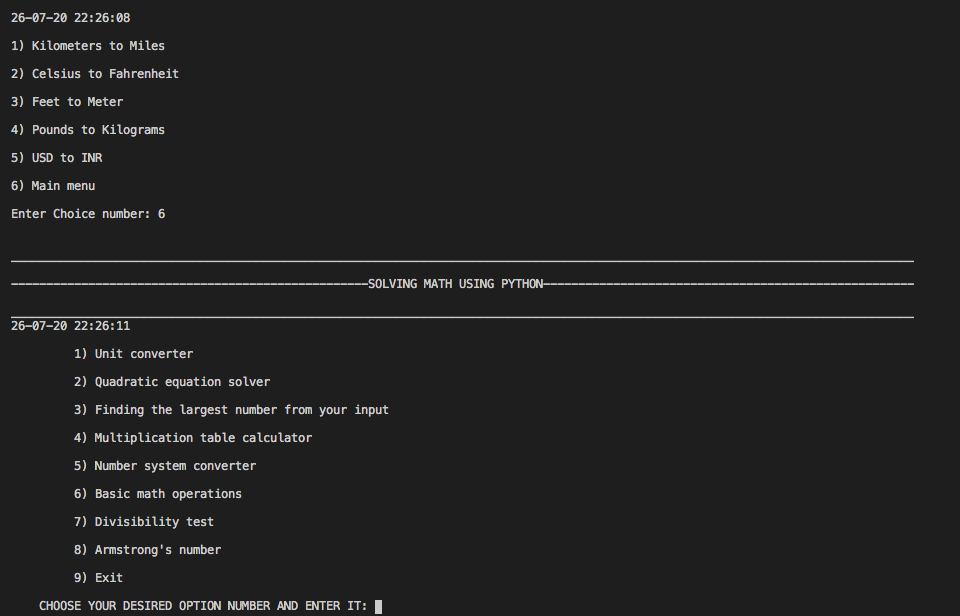
1. Unit convertor:

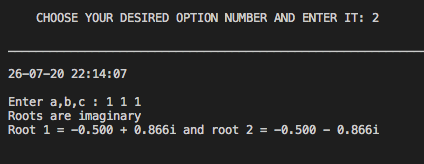
 

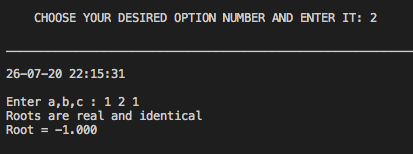


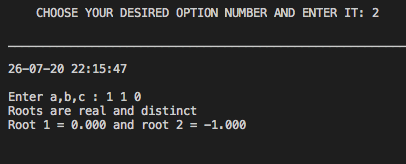


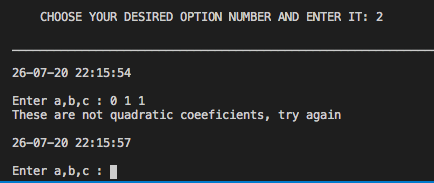


1. Quadratic Equation Solver:

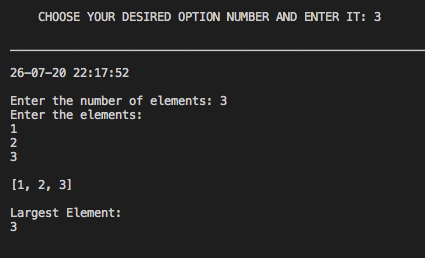
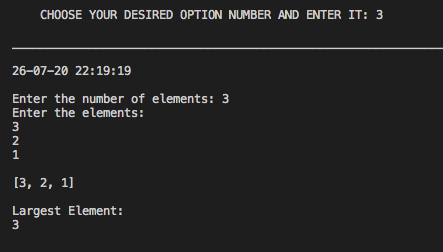


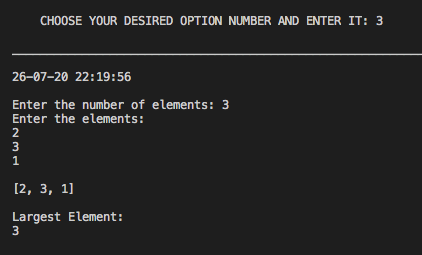
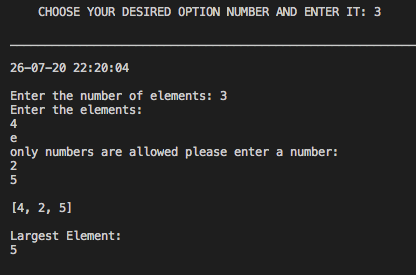




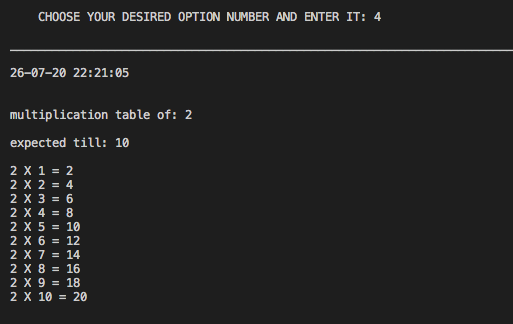


1. Finding the Largest Element:

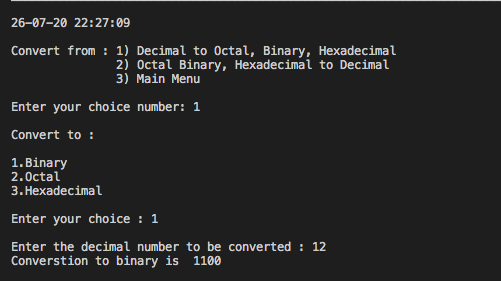
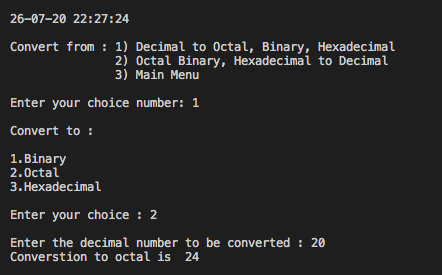
 

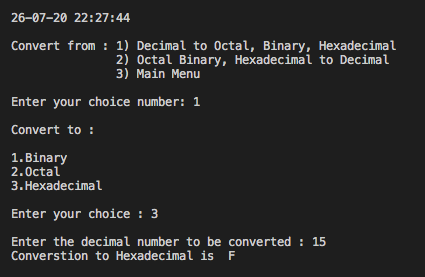
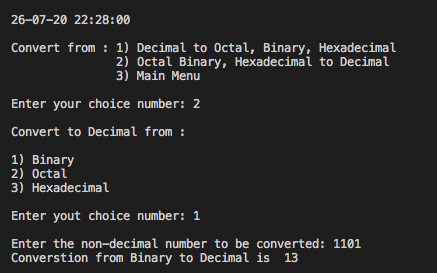
 

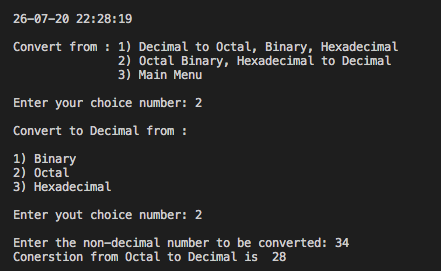
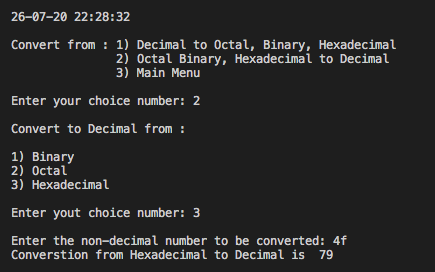
1. Multiplication table calculator:

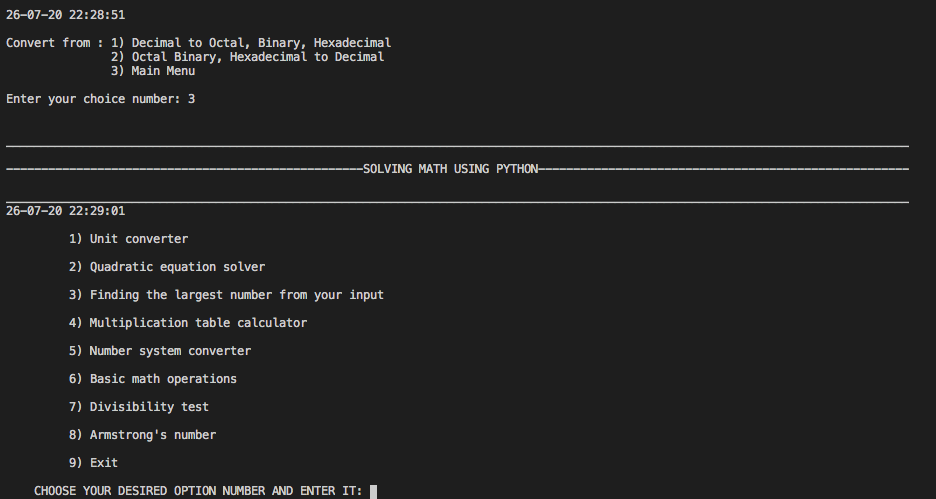


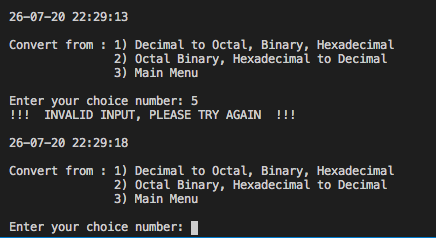
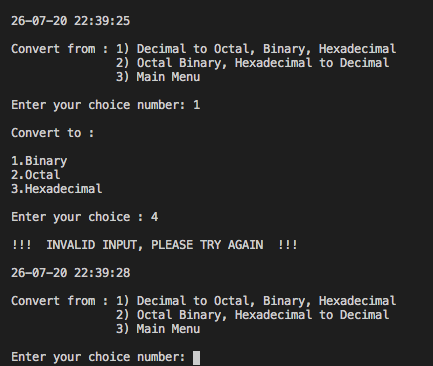
1. Number system converter:

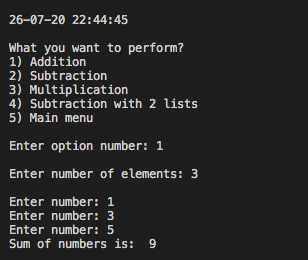
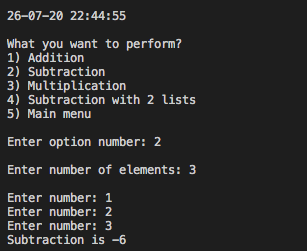
 

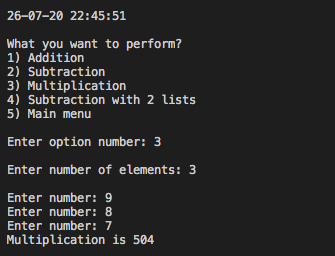
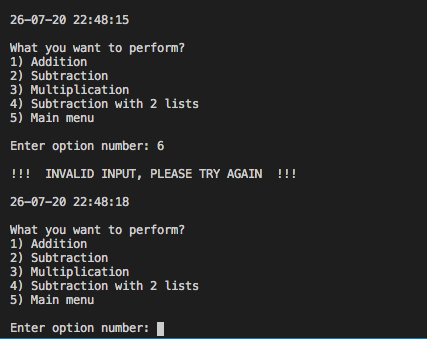
 

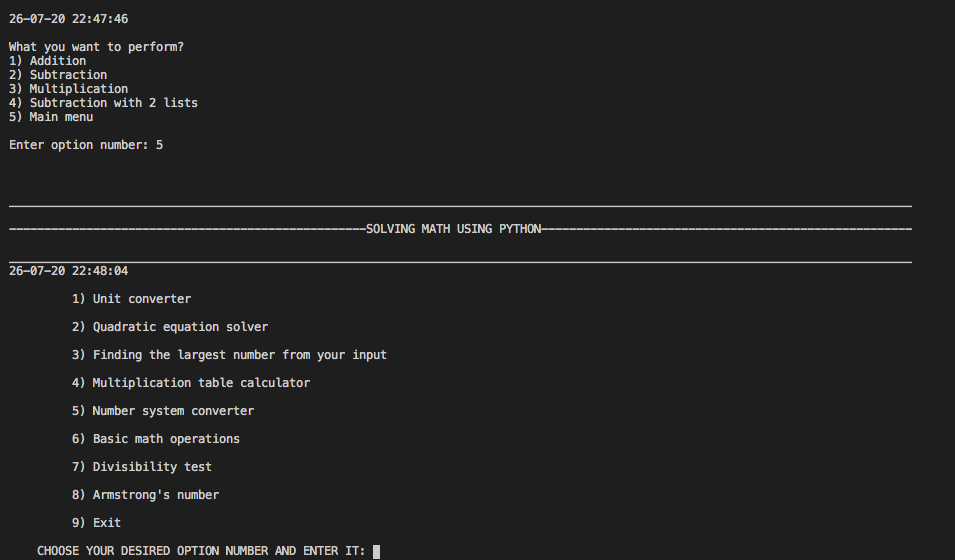


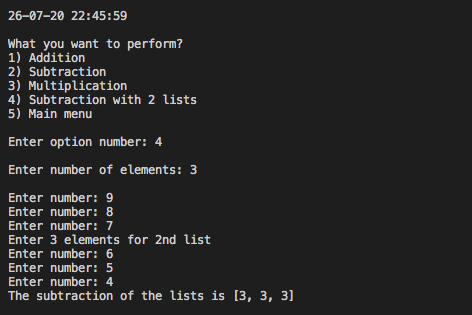
 

1. Basic Math Operations:

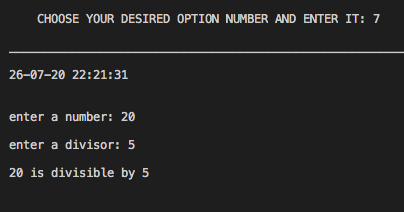
 

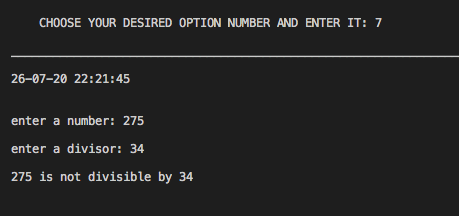
 



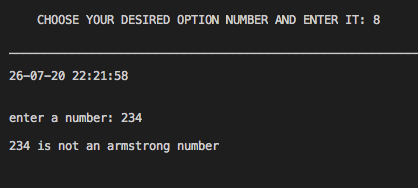


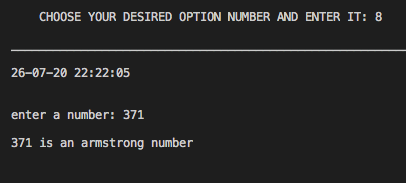
1. Divisibility test:

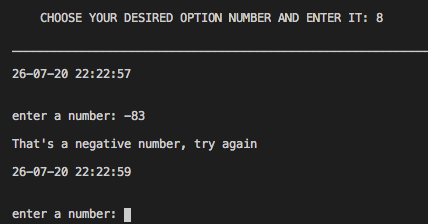




1. Armstrong’s Number:







**Acknowledgment :**

I, on behalf of Group 5, am glad that I got this opportunity to thank our college, APSIT for giving me and my teammates a chance to polish our coding skills. It was great fun to try to code such wonderful programs and test them; we got a lot of insights in our repository of knowledge of coding, learnt a plethora of new essential skills which will be of a great help in our future endeavours.

I along with my teammates would also like to extend our thanks to Prof. Merlin Jacob, our faculty mentor, and Ms. Nimali Keny, our student mentor to help and guide us when things got unclear and tricky. We are fortunate to have such great mentors like them who helped me and my teammates to ace through this project confidently.

Lastly, I would also like to show immense gratitude towards my fellow teammates, Aditya Yadav and Het Patel for working closely and sincerely, and investing their resources to make this project a wonderful success. I learnt a lot through both of them during this course of a week. Thank you.

To conclude, I feel blessed to have had such a great support system for the successful completion of this project, a sincere thanks to all of you.

**References:**

1. Coursera course- [Crash Course on Python from Google](https://www.coursera.org/programs/a-p-shah-institute-of-technology-on-coursera-05d2i?productId=8D3R5HiaEeioIg7r4jw_PA&productType=course&showMiniModal=true)

## Coursera course- [Programming for Everybody (Getting Started with Python)](https://www.coursera.org/programs/a-p-shah-institute-of-technology-on-coursera-05d2i?collectionId=&productId=7A1yFTaREeWWBQrVFXqd1w&productType=course&showMiniModal=true)

1. [Stack overflow](https://stackoverflow.com/questions/49704364/make-python3-as-my-default-python-on-mac/49711594) for making Python 3 as default system interpreter
2. [RealPython](https://realpython.com/python-exceptions/) for try and except exception handling