# Project 1

Mariyah Shahmalak
Instructor: Gary Thai

## *Design:*

- While True
    - Prompt user to choose a type of conversion (or exit) where ‘1’ denotes decimal to hexadecimal conversion, ‘2’ denotes hexadecimal to decimal conversion, ‘3’ denotes decimal to octal conversion, ‘4’ denotes octal to decimal conversion, ‘5’ denotes decimal to binary conversion, ‘6’ denotes binary to decimal conversion, ‘7’ denotes octal to binary conversion, ‘8’ denotes gallons to litres conversion, ‘9’ denotes litres to gallons conversion, and ‘X’ denotes exiting
    - If the user chose ‘1’:
        - Use try-except block to catch Value Error if the user enters any value other than an integer and display an error message; continue the program from beginning
        - If the user enters a number less than zero or greater than 2048 then display an error message; continue the program from beginning
        - Declare a variable ‘one’ and set it equal to the number entered by the user. Declare another variable ‘hexadecimal and initialize it with an empty string
        - While ‘one is greater than 0, append the remainder of that number when divided by 16 to ‘hexadecimal. Then make ‘one equal to itself divided by 16 (using integer division). If the remainder is greater than or equal to 10 then append the corresponding letter
        - Use list slicing to reverse variable ‘hex’
        - Print the converted hexadecimal value
    - Else if the user chose ‘2’:
        - The program prompts the user to enter a hexadecimal number.
        - It then attempts to convert the hexadecimal number to its decimal equivalent.
    - If the input is valid, the conversion is performed using the conversion formula: Adding to the decimal value the result of multiplying the index of the hexadecimal digit in the hexadecimal digits set (hex_digits) by 16 raised to the power of the current position (power)
    - The resulting decimal representation is printed.
    - If the input is not a valid hexadecimal number, a ValueError is caught, and an error message is displayed.
- Else if the user chose ‘3’:
    - The program prompts the user to enter a decimal number.

  - It validates the input to ensure it's within the range of 0 to 2048 and is a valid integer.
  - If the input is valid, the conversion process begins.
  - The decimal number is divided successively by 8, and each remainder is stored.
  - The remainders are then combined in reverse order to form the octal representation.
  - The octal representation is printed as the output.
  - If the input is not valid (e.g., non-integer value or out of range), an error message is displayed.
- Else if the user chose ‘4’:
  - Use try-except block to catch Value Error if the user enters any value other than an octal number and display an error message; continue the program from beginning
  - Declare a variable 'four' and set it equal to the octal number entered by the user
  - Initialize variables 'decimal' and 'power' to 0
  - Use a for loop to iterate through the octal digits in reverse order, multiplying each digit by 8 raised to the power of its position and summing the results
  - Print the original octal number and its decimal representation
- Else if the user chose ‘5’:
  - Prompt user to enter a decimal number between 0 and 2048
  - Use a validate number function to validate the input; if invalid, display an error message and continue the program from beginning
  - Declare a variable 'five' and set it equal to the validated decimal number
  - Initialize a variable 'binary' as an empty string

  Use a while loop to convert the decimal number to its binary representation by repeatedly dividing the number by 2 and prepending the remainder to the binary string
  - Print the original decimal number and its binary representation

- Else if the user chose ‘6’:
  - Prompt user to enter a binary number
  - Use try-except block to catch Value Error if the user enters any value other than a binary number and display an error message; continue the program from beginning
  - Declare a variable 'ix' and set it equal to the binary number entered by the user
  - Initialize variables 'decimal' and 'power' to 0

  Use a for loop to iterate through the binary digits in reverse order, multiplying each digit by 2 raised to the power of its position and summing the results
  - Print the original binary number and its decimal representation

- Else if the user chose ‘7’:
  - Prompt user to enter an octal number
  - Use try-except block to catch ValueError if the user enters any value other than an octal number and display an error message; continue the program from beginning

```
        ▪ Declare a variable  and set it equal to the octal number entered by the user
        ▪ Initialize variables 'decimal' and 'power' to 0
        ▪ Use a for loop to iterate through the octal digits in reverse order, multiplying
          each digit by 8 raised to the power of its position and summing the results
        ▪ Convert the decimal number to its binary representation using a while loop
        ▪ Print the original octal number and its binary representation
▪ Else if the user chose ‘8’:
        ▪  Prompt user to enter a number of gallons
        ▪  Use a validate float function to validate the input; if invalid, display an error
          message and continue the program from beginning
        ▪  Declare a variable 'gallons' and set it equal to the validated input
        ▪ Convert the gallons to Liters using a fixed conversion factor
     Print the original gallons value and its Liters equivalent
▪ Else if the user chose ‘9’:
        ▪ Prompt user to enter a number of Liters
        ▪ Use a validate float function to validate the input; if invalid, display an error
          message and continue the program from beginning
        ▪  Declare a variable Liters and set it equal to the validated input
        ▪  Convert the Liters to gallons using a fixed conversion factor
        ▪  Print the original Liters value and its gallons equivalent
▪ Else
        ▪ Print an error message indicating that the user's choice is invalid; continue the
          program from beginning
```

## *Learning Experience:*

When I first received the assignment instructions, I was confronted with a plethora of unfamiliar terms and concepts, which initially felt daunting. However, rather than feeling overwhelmed, I adopted a systematic approach to tackle the task.

To begin, I focused on comprehensively understanding the concepts outlined in the assignment. Thanks to the resources provided by the professor, such as the instructional YouTube video and accompanying article, I was able to grasp the foundational principles behind hexadecimal, binary, and octal conversions. As someone with a strong affinity for mathematics, these concepts resonated with me, and I found it relatively straightforward to apply mathematical formulas to implement these conversions in code.

Nevertheless, the journey was not without its challenges. I encountered numerous hurdles, including the need to convert string inputs into integer values, a task that required familiarity with data types and conversion methods. Additionally, I learned the significance of employing descriptive variable names, as it enhanced the readability and clarity of my code.

One of the more challenging aspects of the project was implementing input validation to ensure that user-provided values fell within the specified range. This required a thorough understanding of conditional statements and control flow mechanisms, which I acquired through online tutorials and resources like Stack Overflow.

Throughout the project, I leveraged a variety of programming constructs, such as loops (including for loops and while loops), mathematical operators, and conditional statements (if-elif-else), to navigate the complexities of the task. However, one of the most valuable lessons I learned was the importance of meticulous planning and documentation. By outlining the project's functionality and structure in a detailed write-up beforehand, I was able to streamline the coding process and mitigate potential roadblocks.

In conclusion, while the project initially presented challenges due to its unfamiliar terminology and concepts, it ultimately served as a valuable learning experience. By applying programming principles and problem-solving strategies, I was able to overcome obstacles and develop a deeper understanding of numeral systems and coding practices.

## *EXTRA CREDIT/ Enhancement:*

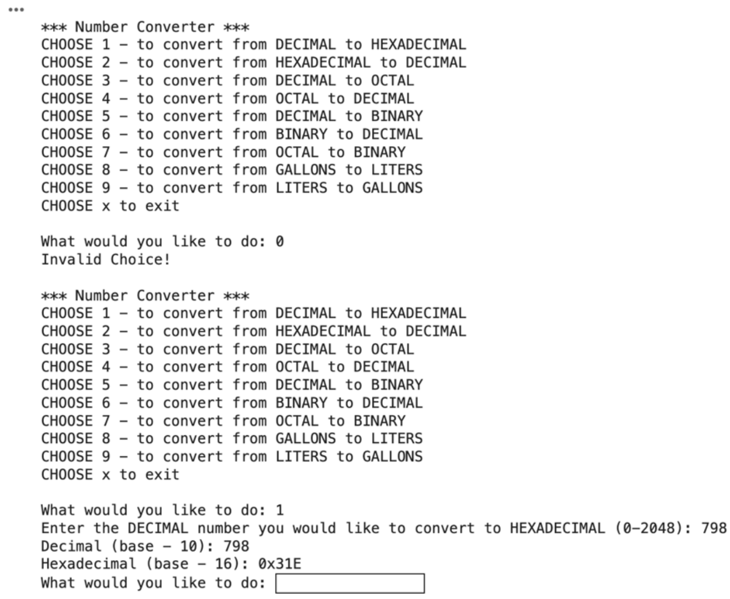
In enhancing my project, I incorporated additional functionality to facilitate conversions between gallons and liters, thus broadening the scope of the application. This involved implementing two new conversion options: gallons to liters and liters to gallons. By seamlessly integrating these conversions into the existing codebase, I expanded the utility of the program, catering to a wider range of user needs. This enhancement not only diversified the conversion capabilities of the application but also showcased my ability to adapt and extend the functionality of the project beyond the initial requirements. Furthermore, the inclusion of these conversions demonstrated my commitment to delivering a comprehensive and user-centric solution, reflecting a proactive approach to software development.

## *Assumptions*

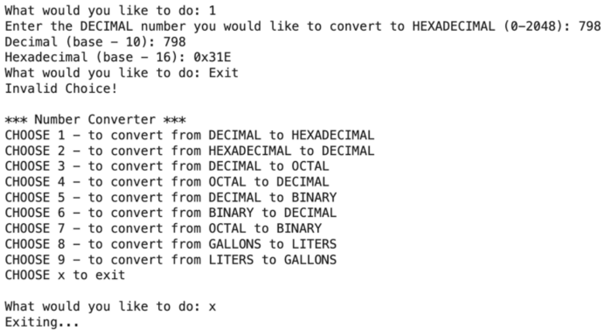
- Users are expected to input valid numeric values for conversions; any non-numeric input will trigger an error message.
- For hexadecimal conversions, the input is assumed to be in the format "0x" followed by valid hexadecimal characters.
- Octal conversions anticipate input starting with a leading zero.
- Binary conversions require input consisting only of 0s and 1s.
- Gallons to Liters and Liters to gallons conversions assume valid numeric input for volume measurements.
- The program provides error handling for invalid inputs, such as strings or out-of-range values.
- All conversions are expected to adhere to the respective mathematical formulas for accurate results.

## *Test Cases:*

Test Case 1



```
...
    *** Number Converter ***
    CHOOSE 1 - to convert from DECIMAL to HEXADECIMAL
    CHOOSE 2 - to convert from HEXADECIMAL to DECIMAL
    CHOOSE 3 - to convert from DECIMAL to OCTAL
    CHOOSE 4 - to convert from OCTAL to DECIMAL
    CHOOSE 5 - to convert from DECIMAL to BINARY
    CHOOSE 6 - to convert from BINARY to DECIMAL
    CHOOSE 7 - to convert from OCTAL to BINARY
    CHOOSE 8 - to convert from GALLONS to LITERS
    CHOOSE 9 - to convert from LITERS to GALLONS
    CHOOSE x to exit

    What would you like to do: 0
    Invalid Choice!

    *** Number Converter ***
    CHOOSE 1 - to convert from DECIMAL to HEXADECIMAL
    CHOOSE 2 - to convert from HEXADECIMAL to DECIMAL
    CHOOSE 3 - to convert from DECIMAL to OCTAL
    CHOOSE 4 - to convert from OCTAL to DECIMAL
    CHOOSE 5 - to convert from DECIMAL to BINARY
    CHOOSE 6 - to convert from BINARY to DECIMAL
    CHOOSE 7 - to convert from OCTAL to BINARY
    CHOOSE 8 - to convert from GALLONS to LITERS
    CHOOSE 9 - to convert from LITERS to GALLONS
    CHOOSE x to exit

    What would you like to do: 1
    Enter the DECIMAL number you would like to convert to HEXADECIMAL (0-2048): 798
    Decimal (base - 10): 798
    Hexadecimal (base - 16): 0x31E
    What would you like to do:
```

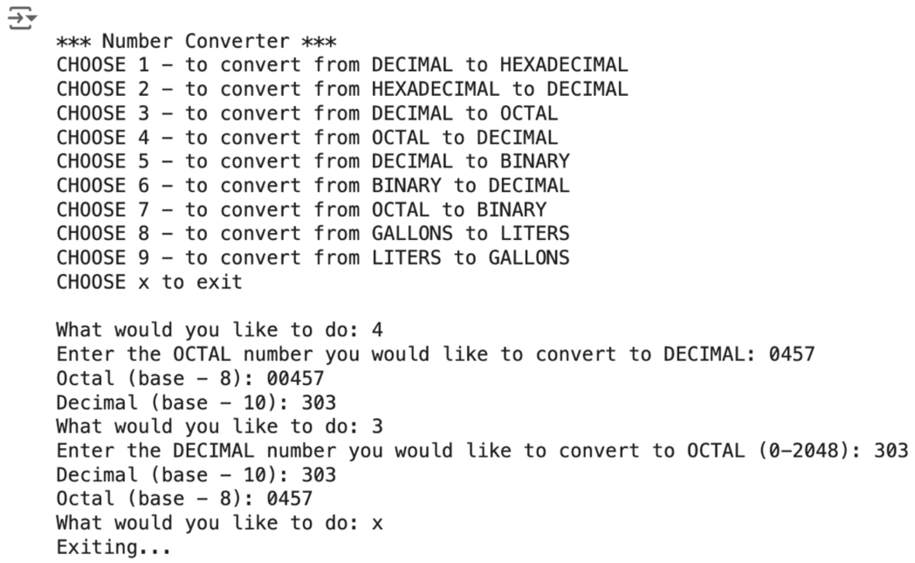


```
What would you like to do: 1
Enter the DECIMAL number you would like to convert to HEXADECIMAL (0-2048): 798
Decimal (base - 10): 798
Hexadecimal (base - 16): 0x31E
What would you like to do: Exit
Invalid Choice!

*** Number Converter ***
CHOOSE 1 - to convert from DECIMAL to HEXADECIMAL
CHOOSE 2 - to convert from HEXADECIMAL to DECIMAL
CHOOSE 3 - to convert from DECIMAL to OCTAL
CHOOSE 4 - to convert from OCTAL to DECIMAL
CHOOSE 5 - to convert from DECIMAL to BINARY
CHOOSE 6 - to convert from BINARY to DECIMAL
CHOOSE 7 - to convert from OCTAL to BINARY
CHOOSE 8 - to convert from GALLONS to LITERS
CHOOSE 9 - to convert from LITERS to GALLONS
CHOOSE x to exit

What would you like to do: x
Exiting...
```

| **Input Type** | **User Input** | **Expected Output** | **Actual Output** | **Pass?** |
|---|---|---|---|---|
| Conversion Option Choice | 0 | Invalid choice | Invalid choice | Yes |
| Conversion Option Choice | 1 | Enter the DECIMAL number you would like to convert to HEXADECIMAL: | Enter the DECIMAL number you would like to convert to HEXADECIMAL: | Yes |
| Decimal number | 798 | Decimal (Base-10): 798, Hexadecimal (Base-16): 0x31E | Decimal (Base-10): 798, Hexadecimal (Base-16): 0x31E | Yes |
| Conversion Option Choice | Exit | Invalid choice | Invalid choice | Yes |

| | | | | |
|---|---|---|---|---|
| Conversion Option Choice | X | | | Yes |

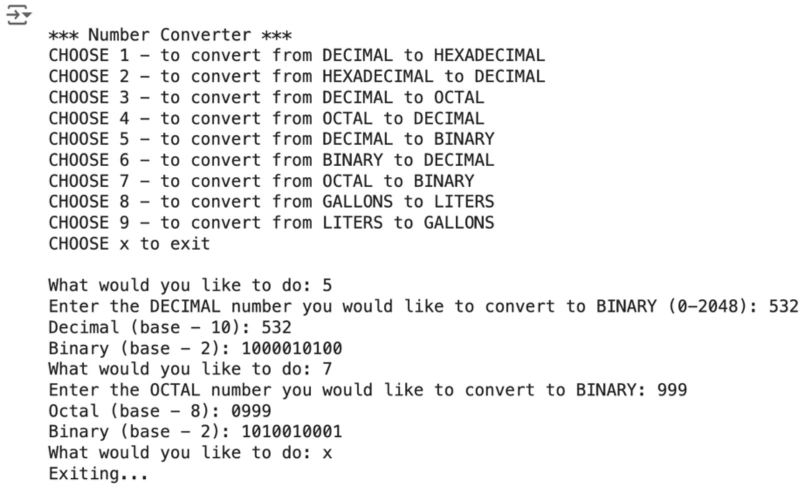
Test Case 2



```
*** Number Converter ***
CHOOSE 1 - to convert from DECIMAL to HEXADECIMAL
CHOOSE 2 - to convert from HEXADECIMAL to DECIMAL
CHOOSE 3 - to convert from DECIMAL to OCTAL
CHOOSE 4 - to convert from OCTAL to DECIMAL
CHOOSE 5 - to convert from DECIMAL to BINARY
CHOOSE 6 - to convert from BINARY to DECIMAL
CHOOSE 7 - to convert from OCTAL to BINARY
CHOOSE 8 - to convert from GALLONS to LITERS
CHOOSE 9 - to convert from LITERS to GALLONS
CHOOSE x to exit

What would you like to do: 4
Enter the OCTAL number you would like to convert to DECIMAL: 0457
Octal (base - 8): 00457
Decimal (base - 10): 303
What would you like to do: 3
Enter the DECIMAL number you would like to convert to OCTAL (0-2048): 303
Decimal (base - 10): 303
Octal (base - 8): 0457
What would you like to do: x
Exiting...
```

| **Input Type** | **User Input** | **Expected Output** | **Actual Output** | **Pass?** |
|---|---|---|---|---|
| Conversion Option Choice | 4 | Enter the OCTAL number you would like to convert to DECIMAL: | Enter the OCTAL number you would like to convert to DECIMAL: | Yes |
| Octal Number | 0457 | Octal (Base-8): 0457, Decimal (Base-10): 303 | Octal (Base-8): 0457, Decimal (Base-10): 303 | Yes |
| Conversion Option Choice | 3 | Enter the DECIMAL number you would like to convert to OCTAL: | Enter the DECIMAL number you would like to convert to OCTAL: | Yes |
| Decimal Number | 303 | Decimal (Base-10): 303, Octal (Base-8): 0457 | Decimal (Base-10): 303, Octal (Base-8): 0457 | Yes |
| Conversion Option Choice | X | | | Yes |

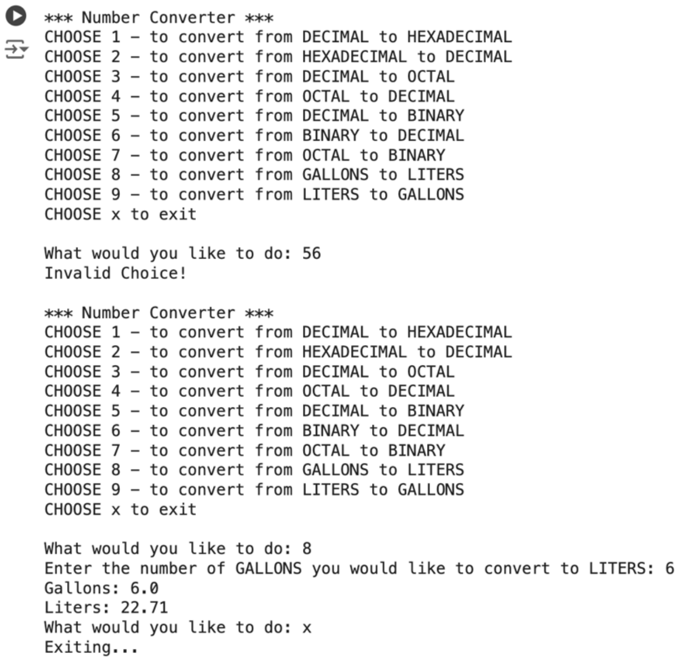
Test Case 3:



```
*** Number Converter ***
CHOOSE 1 - to convert from DECIMAL to HEXADECIMAL
CHOOSE 2 - to convert from HEXADECIMAL to DECIMAL
CHOOSE 3 - to convert from DECIMAL to OCTAL
CHOOSE 4 - to convert from OCTAL to DECIMAL
CHOOSE 5 - to convert from DECIMAL to BINARY
CHOOSE 6 - to convert from BINARY to DECIMAL
CHOOSE 7 - to convert from OCTAL to BINARY
CHOOSE 8 - to convert from GALLONS to LITERS
CHOOSE 9 - to convert from LITERS to GALLONS
CHOOSE x to exit

What would you like to do: 5
Enter the DECIMAL number you would like to convert to BINARY (0-2048): 532
Decimal (base - 10): 532
Binary (base - 2): 1000010100
What would you like to do: 7
Enter the OCTAL number you would like to convert to BINARY: 999
Octal (base - 8): 0999
Binary (base - 2): 1010010001
What would you like to do: x
Exiting...
```

| **Input Type** | **User Input** | **Expected Output** | **Actual Output** | **Pass?** |
|---|---|---|---|---|
| Conversion Option Choice | 5 | Enter the DECIMAL number you would like to convert to BINARY (0-2048): | Enter the DECIMAL number you would like to convert to BINARY (0-2048): | Yes |
| Decimal Number | 532 | Decimal (base - 10): 532<br>Binary (base - 2): 1000010100 | Decimal (base - 10): 532<br>Binary (base - 2): 1000010100 | Yes |
| Conversion Option Choice | 7 | Enter the OCTAL number you would like to convert to BINARY: | Enter the OCTAL number you would like to convert to BINARY: | Yes |
| Octal Number | 999 | Octal (base - 8): 0999<br>Binary (base - 2): 1010010001 | Octal (base - 8): 0999<br>Binary (base - 2): 1010010001 | Yes |
| Conversion Option Choice | X | Exiting... | Exiting... | Yes |

Test Case 4:



```
*** Number Converter ***
CHOOSE 1 - to convert from DECIMAL to HEXADECIMAL
CHOOSE 2 - to convert from HEXADECIMAL to DECIMAL
CHOOSE 3 - to convert from DECIMAL to OCTAL
CHOOSE 4 - to convert from OCTAL to DECIMAL
CHOOSE 5 - to convert from DECIMAL to BINARY
CHOOSE 6 - to convert from BINARY to DECIMAL
CHOOSE 7 - to convert from OCTAL to BINARY
CHOOSE 8 - to convert from GALLONS to LITERS
CHOOSE 9 - to convert from LITERS to GALLONS
CHOOSE x to exit

What would you like to do: 56
Invalid Choice!

*** Number Converter ***
CHOOSE 1 - to convert from DECIMAL to HEXADECIMAL
CHOOSE 2 - to convert from HEXADECIMAL to DECIMAL
CHOOSE 3 - to convert from DECIMAL to OCTAL
CHOOSE 4 - to convert from OCTAL to DECIMAL
CHOOSE 5 - to convert from DECIMAL to BINARY
CHOOSE 6 - to convert from BINARY to DECIMAL
CHOOSE 7 - to convert from OCTAL to BINARY
CHOOSE 8 - to convert from GALLONS to LITERS
CHOOSE 9 - to convert from LITERS to GALLONS
CHOOSE x to exit

What would you like to do: 8
Enter the number of GALLONS you would like to convert to LITERS: 6
Gallons: 6.0
Liters: 22.71
What would you like to do: x
Exiting...
```

| Input Type | User Input | Expected Output | Actual Output | Pass? |
|---|---|---|---|---|
| Conversion Option Choice | 56 | Invalid Choice! | Invalid Choice! | Yes |
| Conversion Option Choice | 8 | Enter the number of GALLONS you would like to convert to LITERS: | Enter the number of GALLONS you would like to convert to LITERS: | Yes |
| Gallons - Litres | 6 | Liters: 22.71 | Liters: 22.71 | Yes |
| Conversion Option Choice | X | Exiting... | Exiting... | Yes |