Final Submission Write-Up

1. Project Design

* While True:
* Prompt user to choose a type of conversion or exit:
  + ‘1’ converts decimal to hexadecimal conversion,
  + ‘2’ converts decimal to octal conversion,
  + ‘3’ converts decimal to binary conversion,
  + ‘4’ converts binary to decimal, and
  + ‘X’ exits the program:
* Set user choice flag, ‘user\_choice’ to -1
* Set user input flag, ‘user\_input’ to -1
* While user choice flag, ‘‘user\_choice” is -1:
* If user chose ‘X’, then exit the program by using exit()
* If user chose ‘1’, ‘2’, ‘3’ or ‘4’ , then prompt user to enter a decimal number to be converted to the appropriate number type
* While the entered number is out of range, ask the user to enter again
  + Try/Except when the entry is not an integer, prompt the user to re-enter an integer again
  + If the entered number is out of range, ask the user to enter a new number
  + Else, convert the number of the user choice by calling the “decimal\_to\_hex()” , “decimal\_oct()”, “decimal\_to\_binary()” , “binary\_to\_decimal()” respectively.
* Else, the user choice is out of range 1-4, prompt the user to re-choose an option from the given options
* End While

def decimal\_to\_hex(decimal):

* If the user chose ‘1’, then do hexadecimal conversion: initialize an empty list and an empty var “result” to be returned
* While the decimal is not 0, append the remainder of that number when divided by 16 to the list. Then make that number equal to itself divided by 16 (using integer division).
* Reverse the elements of the list by using the [list].reverse()
* Convert the int list to a string list using a for loop
* Iterate through the list and set each number greater than or equal to 10 equals to its corresponding hexadecimal alphabetical value.
* Then iterate though the list and append each element to an empty string.
* Finally, return the hexadecimal number

def decimal\_to\_oct(decimal):

* If the user chose ‘2’, then do octal conversion: initialize an empty list and an empty var “result” to be returned
* While the decimal is not 0, append the remainder of that number when divided by 16 to the list. Then make that number equal to itself divided by 16 (using integer division).
* Reverse the elements of the list by using the [list].reverse()
* Convert the int list to a string list using a for loop
* Iterate through the list and set each number greater than or equal to 10 equals to its corresponding hexadecimal alphabetical value.
* Then iterate though the list and append each element to an empty string.
* Finally, return the hexadecimal number

def decimal\_to\_binary(decimal):

* If the user chose ‘3’, then do binary conversion: initialize an empty list and an empty var “result” to be returned
* While the decimal is not 0, append the remainder of that number when divided by 2 to the list. Then make that number equal to itself divided by 2 (using integer division).
* Reverse the elements of the list by using the [list].reverse()
* Convert the int list to a string list using a for loop
* Then iterate though the list and append each element to an empty string.
* Finally, return the hexadecimal number

def binary\_to\_decimal(binary):

* If the user chose ‘4’, then do binary to decimal conversion: initialize a variable ‘power’ to 0 and an empty int “decimal\_number” to be returned
* While the binary number is greater than 0
* Take the first right digit of the binary number by using modulus ( %10 )
* Multiply the digit by ( 2 to ‘power’)
* Add the value to decimal\_number
* Add ‘power’ +1
* Divide the ‘decimal\_number’ by 10 (using integer divison)
* Loop until the binary number becomes 0

2. Test Cases

Test Case 1

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: -1

Wrong number, enter a number between 1-4 or X to exit

--------------------------------------------------------

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: a

Wrong number, enter a number between 1-4 or X to exit

--------------------------------------------------------

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: 3

Enter a decimal number: 257

Please. Choose a number between 0 and 256

Enter a decimal number: 256

Base-10: 256

Binary: 100000000

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: x

Thanks for using Converter Number Program

Program is closing...

PS C:\Users\pjose\Documents\CMSC 206 (Python)\Project>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input Type** | **User Input** | **Expected Output** | **Actual Output** | **Pass?** |
| Conversion Option Choice | -1 | Wrong number, enter a number between 1-4 or X to exit | Wrong number, enter a number between 1-4 or X to exit | Yes |
| Conversion Option Choice | a | Wrong number, enter a number between 1-4 or X to exit | Wrong number, enter a number between 1-4 or X to exit | Yes |
| Decimal number | 257 | Please. Choose a number between 0 and 256 | Please. Choose a number between 0 and 256 | Yes |
| Decimal number | 256 | Base-10: 256  Binary: 100000000 | Base-10: 256  Binary: 100000000 | Yes |
| Conversion Option Choice | x | Thanks for using Converter Number Program  Program is closing... | Thanks for using Converter Number Program  Program is closing... | Yes |

Test Case 2:

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: 4

Enter a binary number: 1002

Please. Choose a number with 0s and 1s

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: X

Thanks for using Converter Number Program

Program is closing...

PS C:\Users\pjose\Documents\CMSC 206 (Python)\Project> & C:/Users/pjose/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/pjose/Documents/CMSC 206 (Python)/Project/Valdivia\_Project\_1.py"

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: 4

Enter a binary number: 101000

Binary: 101000

Base-10: 40

----------------------------------------------------------------

Number Converter: Convert a number (between 0 and 1024) from

Enter 1 -- Decimal to Hexadecimal

Enter 2 -- Decimal to Octal

Enter 3 -- Decimal to Binary (Between 0 and 256)

Enter 4 -- Binary to Decimal (only 1 and 0)

Enter X -- Exit

Enter your choice: 2

Enter a decimal number: 1200

Please. Choose a number between 0 and 1024

Enter a decimal number: 870

Base-10: 870

Octal: 01546

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input Type** | **User Input** | **Expected Output** | **Actual Output** | **Pass?** |
| Conversion Option Choice | 4 | Enter a binary number: | Enter a binary number: | Yes |
| Decimal number | 1002 | Please. Choose a number with 0s and 1s | Please. Choose a number with 0s and 1s | Yes |
| Conversion Option Choice | X | Choose one of the options given: | Choose one of the options given: | Yes |
| Conversion Option Choice | 4 | Enter a binary number: | Enter a binary number: |  |
| Binary number | 101000 | Binary: 101000  Base-10: 40 | Binary: 101000  Base-10: 40 | Yes |
| Conversion Option Choice | 2 | Enter a decimal number: | Enter a decimal number: |  |
| Decimal number | 1200 | Please. Choose a number between 0 and 1024 | Please. Choose a number between 0 and 1024 | Yes |
| Decimal number | 870 | Base-10: 870  Octal: 01546 | Base-10: 870  Octal: 01546 | Yes |

3. Learning Experience

This project was a great assignment and warm-up introduction to python. The assignment was about converting numbers in different conversions, such as decimal, hexadecimal, octal, and binary numbers. I was unfamiliar in how to convert to these kinds of numbers, which complicate me at first point, but it was no impossible to code. So, I went to many videos resources to learn how to accurate convert between these format numbers. While I was watching the videos of converting numbers, I learned the process and coded the algorithm to convert decimal numbers to any number type. At the beginning of the project, I first started to code the menu which took me a while to make it done. Because I am proficient in Java language, I could implement some ideas and data structures in order to make the program functionate well. However, there were some complications with Python. One of these, the do-while loop statement is not integrated in Python, which is helpful when creating menus, so I had to use another resource such as a while loop. Also, initializing variables was one of the issues that costed me time. In Java, it was possible to initialize variables without using it, but there is no such way in Python. For example, for lists, I needed to initialize and declare like  . Another example, for string variables, I needed to initialize and declare like . If I don’t do this, it will give me some issues saying, “variable is not defined.” Consequently, I searched many articles to learn how to solve these kinds of problems and finally I could understand better about Python syntaxes and rules. Furthermore, the main objective of this assignment is to program a number converter, but I knew that each conversion would have their own algorithm, so I decided to use function to group each algorithm for each method conversion, and since I always used methods from my Java experience to make the code cleaner, I knew that function will really help me to make the program code cleaner and understandable for other programmers.

Moreover, I also learned how to implement try-except blocks in the program, which helped me to validate users’ inputs. In my opinion, the most valuable knowledge I learned from this assignment was how to use for loops, lists, sets, and strings perfectly, even some of those concepts are not taught yet. Specifically, I learned how to iterate through a list either from left to right, or right to left, how to reverse a list by simply using reverse() function, how to create and use a set for the binary number validation, how to change the data type of a list by simply using one line of for loop code.

At the end, the program works very well and has not complications with user-program interaction. Perhaps, I could have improved the menu of the program since it loops forever until the user press ‘X’ or ‘x’ by doing a do-while loop. Overall, it was a fun program to code, and I am ready for any upcoming challenges

4. Assumptions

* The user might enter a choice other than the choices listed. However, if the user chooses a no listed option, the user will be asked to choose again
* The user might enter a string or char value when asked to choose a number. However, if the user chooses a string or char, the program will ask the user to re-enter a new number
* The program will be executed over and over. However, the user may choose to exit the program by choosing the ‘X’ or ‘x’
* If the user enters an invalid binary number format, the program will ask the user to enter a valid number