曾郁珊 D1265154 Assignment4 report

Challenges Faced:

Hexadecimal addition:

1. Handling variable string lengths:

Challenge: Managing cases where the lengths of the two input hexadecimal strings differed.

Solution: Implemented conditional checks to identify the longer string and adjusted the addition logic accordingly, considering the positional gap between the two strings.

2. String conversion to decimal:

Challenge: Converting a hexadecimal string to its decimal equivalent posed challenges in calculating the positional weight for each digit.

Solution: Developed a separate function, strConvertDec, to calculate the decimal value of a hexadecimal string. The function used nested loops to iterate through each digit and compute its decimal value based on positional weight.

3. Formatting and printing:

Challenge: Ensuring proper formatting and alignment of the output for readability.

Solution: Utilized the printf function with formatting specifiers to align the input numbers and result in a visually clear manner. Calculated the width dynamically based on the length of the result.

4. Input validation:

Challenge: Ensuring that the entered numbers were valid hexadecimal representations.

Solution: Implemented a check function to validate the input strings, displaying an error message and prompting for re-entry in case of invalid input.

5. Handling carry values:

Challenge: Efficiently managing carry values during addition, especially when dealing with the rightmost digits.

Solution: Employed conditional checks to determine if a carry was needed, updating the carry variable accordingly. Special attention was given to the rightmost digit to prevent array index out-of-bounds errors.

Hexadecimal multiplication:

1. Multiplying Hexadecimal Digits:

Challenge: Performing multiplication of individual hexadecimal digits while considering their positional weights.

Solution: Developed a nested loop structure to multiply each digit of one number with every digit of the other, considering their positional weights and summing the results.

2. Carry Management:

Challenge: Efficiently managing carry values during the multiplication process.

Solution: Employed conditional checks to determine if a carry was needed, updating the carry variable accordingly. Special attention was given to handling carry overflow in the final result.

3. Power of 16 Computation:

Challenge: Calculating powers of 16 for positional weights during the conversion of a hexadecimal string to its decimal equivalent.

Solution: Developed a separate function, ex, to compute the power of 16. This function was utilized within the strConvertDec function to calculate positional weights dynamically.