**Project 1**

In this project, we are requested to build a program (Arithmetic Calculator) that can multiply any 2 positive numbers with the same base of any length.

For this, we need to develop an algorithm that can implement the multiplication process with fastest time and the least memory space used through the implementation of this program.

Three main stages are involved in creating an algorithm: data input, data processing, and results output.

Through this report, we will start to develop an efficient algorithm for this problem.

Stages: -

* Input file (obtain data).
* Data Processing (Multiplication operation).
* Output file (Results).

Requirements: -

* 2 positive numbers (can be infinite).
* Base of the system (between 2 and 10 both included otherwise is considered as an error).
* Conversion to base 10
* Result displayed in the base of the system and base 10.

Challenges: -

* Organizing the input (store data using the least possible memory spaces).
* Input arithmetic numbers can be infinite numbers which can make multiplication very complex and time consuming.
* Different arithmetic bases require different multiplication rules and different carry out values.
* Applying Multiplication operation for arithmetic numbers in the most sufficient way and the least time consuming to minimize the execution time as possible.
* Displaying output in an efficient way (system base and base 10).

Solutions: -

Approach (1): -



* For taking input (Linked Lists) will be used to be able to handle the infinite numbers and their multiplications.
* These operations will be different from base to base (2 to 10 both included).
* This approach will give as a time complexity of Big O(n) and a shorter execution time.
* Ex:

In the input.txt (?13245262 X 53653563 -------------)

These digits will be stored in different nodes in the linked list. We will store either 1 digit in each and apply operations on them depending on the base of the system. The (?) represents the first node that stores the base value of the arithmetic system.

The base 10 multiplication will be placed under the

Grade-school multiplication (Standard-Algorithm) will be done digit by digit and will be added to the result considering the carry out values.

The value of the carry out will depend on the base value of the system and the digits multiplication one by one.

* After the multiplication operations are finished, we will obtain the results.
* The results will be displayed in the output file using file processing management.

Approach (2): -

Shape

Description automatically generated with low confidence

* This is a sample of the input file. For each number in this case, we will create a linked list that ends with the base number. (The node before the base number will store a free space).
* After the end of the first number, we keep checking the line till we reach a number and store it in the same way.
* So basically 2 linked lists per line.
* Then we make sure that the 2 numbers in each line are the same by converting the first number to base 10 and check if it is the same to the second number.
* Multiplication (first & third) & (second & forth) linked lists will be multiplied digit by digit using the standard algorithm for multiplication.
* Split the number into digits in a doubly linked list. Using basic addition principles that goes digit by digit, with a carry are implemented in add and subtract functions.
* These functions are now used to carry out the multiplication and division operations using the basic approach of multiplying last digit to all and then shifting and adding or finding the closest large multiple to divisor to divide the dividend.
* After the multiplication operations are finished, we will obtain the results.
* The results will be displayed in an output file using file processing operators.

This is an initial approach to the solution; these algorithms will be tested and may be modified later depending on the results of the conducted tests to examine their efficiency.