ADDITION (b, c, k) (1)

‘’’Parameters are b and c and are arrays. Input K is an integer which indicates after how many numbers entered into system, a new row is started. Returns and thus calls Format function which has the array b which has had c added to it and the integer k passed to it. So that the matrix can be formatted for output.‘’’

For i <- 1 to length[b] (n)

b[i] <- b[i] + c[i] (n)

return Format(b,k) (1)

SUBTRACTION (b, c, k) (1)

‘’’Parameters are b and c and are arrays. Input K is an integer which indicates after how many numbers entered into system, a new row is started. Returns and thus calls Format function which has the the array b which has had c taken away from it and the integer k passed to it. So that the matrix can be formatted for output’’’

For i <- 1 to length[b] (n)

b[i] <- b[i] – c[i] (n)

return Format(b, k) (1)

MULTIPLY (b, c) (1)

‘’’ two inputs (b, c) of two matrices with their structure being b(row, column). It will return the matrix produced by multiplying b and c together. The first for loop goes through the number of rows and the second through the number of columns of b. As the number of rows of the first matrix has to be equal to the number of columns of the second for multiplication, by switching the identifiers of I and j around, it will therefore multiply the numbers in the row by the numbers in the column. For each row and column.’’’

For i <- 1 to numberOfRows[b] (n)

For J <-1 to numberOfColumns[b] (n^2)

numberHolder <- numberHolder + b[i,J] multiplied by c[J,i] (n^2)

multipliedMatrix[i,J] <- numberHolder (n)

numberHolder <- 0 (n)

return multipliedMatrix (1)

Run time of A:

2\*(B+C) = 4n + 2

B\*C = 2n^2 + 3n + 2

B\*C-2\*(B+C) = 2n + 2

Total run-time:

2n^2+9n+6

O(n^2)