

Name	Siddhesh Sonar
UID no.	2021700063
Experiment No.	5

AIM:	To implement Matrix Chain Multiplication
Program	
PROBLEM STATEMENT :	To implement Matrix Chain Multiplication
ALGORITHM/ THEORY:	<p>Matrix chain multiplication is an optimization problem concerning the most efficient way to multiply a given sequence of matrices. The problem is not actually to perform the multiplications, but merely to decide the sequence of the matrix multiplications involved.</p> <pre> Begin define table minMul of size n x n, initially fill with all 0s for length := 2 to n, do for i:=1 to n-length, do j := i + length - 1 minMul[i, j] := ∞ for k := i to j-1, do q := minMul[i, k] + minMul[k+1, j] + array[i-1]*array[k]*array[j] if q < minMul[i, j], then minMul[i, j] := q done done done return minMul[1, n-1] End </pre>

PROGRAM:

```
#include <limits.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void print_parentheses(int s[][5], int i, int j) {
    if (i == j) {
        printf("A%d", i);
        return;
    }
    printf("(");
    print_parentheses(s, i, s[i][j]);
    print_parentheses(s, s[i][j] + 1, j);
    printf(")");
}

int matrixmin(int p[], int n) {
    int m[n][n];
    int s[n][n];
    memset(m, 0, sizeof(m[0][0]) * n * n);

    int i, j, k, L, q;

    for (L = 2; L < n; L++) {
        for (i = 1; i < n - L + 1; i++) {
            j = i + L - 1;
            m[i][j] = INT_MAX;
            for (k = i; k <= j - 1; k++) {
                q = m[i][k] + m[k + 1][j] + p[i - 1] * p[k] *
p[j];
                if (q < m[i][j]) {
                    m[i][j] = q;
                    s[i][j] = k;
                }
            }
        }
    }

    printf("\nm Table:\n");
    for (i = 1; i < n; i++) {
        for (j = 1; j < n; j++) {
            printf("%d ", m[i][j]);
        }
        printf("\n");
    }
}
```

```

    }

    printf("\ns Table:\n");
    for (i = 1; i < n - 1; i++) {
        for (j = 2; j < n; j++) {
            if (i < j) {
                printf("%d ", s[i][j]);
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }

    printf("\nOptimal Multiplication: ");
    print_parentheses(s, 1, n - 1);
    printf("\n");

    return m[1][n - 1];
}

int main() {
    int arr[] = {1, 2, 3, 2, 1};
    int size = sizeof(arr) / sizeof(arr[0]);

    printf("\nMinimum cost = %d\n", matrixmin(arr, size));

    return 0;
}

```

RESULT:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  SQL CONSOLE  COMMENTS  TERMINAL

Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.

C:\Siddhesh\Github\DAA>cd "c:\Siddhesh\Github\DAA\DAA_Exp_5\" && gcc matrix_chain.

m Table:
0 6 12 14
0 0 12 12
0 0 0 6
0 0 0 0

s Table:
1 2 1
  2 2
    3

Optimal Multiplication: (A1(A2(A3A4)))

Minimum cost = 14

c:\Siddhesh\Github\DAA\DAA_Exp_5>
```

CONCLUSION:

Successfully understood Matrix Chain Multiplication algorithm and implemented it in C program to find the minimum number of steps for multiplication of 2 matrices.