

Assignment#6 - Matrix Chain Multiplication Problem Implementation using Dynamic Programming in any of your preferred Programming Language (C/C++/Java)

Code:

```
#include<bits/stdc++.h>

using namespace std;

int MAX = 99999999;

int MatrixChain(int p[], int n)
{
    int mat [n][n];

    int j,m;

    for (int i=1; i<n; i++)
    {
        mat[i][i] = 0;

        for (int l=2; l<n; l++)
        {
            for (int i=1; i<n-l+1; i++)
            {
                j=i+l-1;

                mat[i][j] = MAX;
```

```

        for (int k=i; k<=j-1; k++)
        {
            m = mat[i][k] + mat[k+1][j] + p[i-1]*p[k]*p[j];
            if (m<mat[i][j])
            {
                mat[i][j]=m;
            }
        }
    }
}

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

```

```

int main ()
{
    int n,i;

    cout << "Enter number of matrices = ";

    cin >> n;

    int arr[n];

    cout << endl << "Enter dimensions = " << endl;

    for (i=0; i<n; i++)
    {
        cout << "Enter d" << i << " = ";

        cin >> arr[i];
    }
}

```

```

}

int size = sizeof (arr)/sizeof (arr[0]);

cout << "Minimum number of multiplications is = " << MatrixChain(arr, size);

return 0;

}

```

Output:

The screenshot shows a C++ IDE with the following code in `Matrix chain multiplication.cpp`:

```

1  #include<bits/stdc++.h>
2  using namespace std;
3
4  int MAX = 99999999;
5  int MatrixChain(int p[], int n)
6  {
7      int mat [n][n];
8      int j,m;
9      for (int i=1; i<n; i++)
10     {
11         mat[i][i] = 0;
12         for (int l=2; l<n; l++)
13         {
14             for (int i=1; i<n-l+1; i++)
15             {
16                 j=i+l-1;
17                 mat[i][j] = MAX;
18                 for (int k=i; k<=j-1; k++)
19                 {
20                     m = mat[i][k] + mat[k+1][j] + p[i-1]*p[k]*p[j];
21                     if (m<mat[i][j])
22                     {
23                         mat[i][j]=m;
24                     }
25                 }
26             }
27         }
28     }
29 }

```

The output window shows the following execution results:

```

Enter number of matrices = 5
Enter dimensions =
Enter d0 = 2
Enter d1 = 3
Enter d2 = 4
Enter d3 = 5
Enter d4 = 3
Minimum number of multiplications is = 94
Process returned 0 (0x0)   execution time : 4.906 s
Press any key to continue.

```