MD NAHIDUL HOQUE NIRAB 2018100000023

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:

