```
def matrixChainOrder(p):
    INTMAX = 9999999999
    n = len(p)
    M = [[0 for i in range(0,n)] for i in range(0,n)]
    S = [[0 for i in range(0,n)] for i in range(0,n)]
    for i in range(1,n):
        M[i][i] = 0

    for l in range(2,n):
        for i in range(1,n-l+1):
            j = i+l -1
            M[i][j]= INTMAX
            for k in range(i,j):
                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
    #print M
    return M[1][n-1],S

import sys

def printOrder(S,i,j):
    if i==j:
        sys.stdout.write("A[")
        sys.stdout.write(str(i))
        sys.stdout.write("]")
    else:
        sys.stdout.write("(")
        printOrder(S,i,S[i][j])
        printOrder(S,S[i][j]+1,j)
        sys.stdout.write(")")

print "Number of input"
N = input()
print "Enter sizes of matrix:: "
a = [int(x) for x in raw_input().strip().split()]
from time import time
t0 = time()
M,S = matrixChainOrder(a)
print "Min Multiplication required :: ", M
print "Order of Multiplication ::"
printOrder(S,1,len(a)-1)
print ""
```