#include<bits/stdc++.h>

using namespace std;

int solve\_dp(vector<int> v, int i, int j, vector<vector<int> > &dp){

//base case

if(i>=j) return dp[i][j] = 0;

// actual case

if(dp[i][j]==-1){

dp[i][j] = INT\_MAX;

for(int k=i;k<=j-1;k++){

int temp\_ans = solve\_dp(v,i,k,dp) + solve\_dp(v,k+1,j,dp) + v[i-1]\*v[k]\*v[j];

dp[i][j] = min(temp\_ans, dp[i][j]);

}

}

return dp[i][j];

}

int solve\_rec(vector<int> v, int i, int j){

if(i>=j){

return 0;

}

int min\_num = INT\_MAX;

for(int k=i;k<=j-1;k++){

int temp\_ans = solve\_rec(v,i,k) + solve\_rec(v,k+1,j) + v[i-1]\*v[k]\*v[j];

min\_num = min(min\_num,temp\_ans);

}

return min\_num;

}

int main(){

int n;

cin>>n;

vector<int> v(n);

for(int i=0;i<n;i++){

int x;

cin>>x;

v[i] = x;

}

vector<vector<int> > dp(n+1,vector<int> (n+1,-1));

cout<<solve\_dp(v,1,n-1,dp);

}