DESIGN AND ANALYSIS OF ALGORITHMS

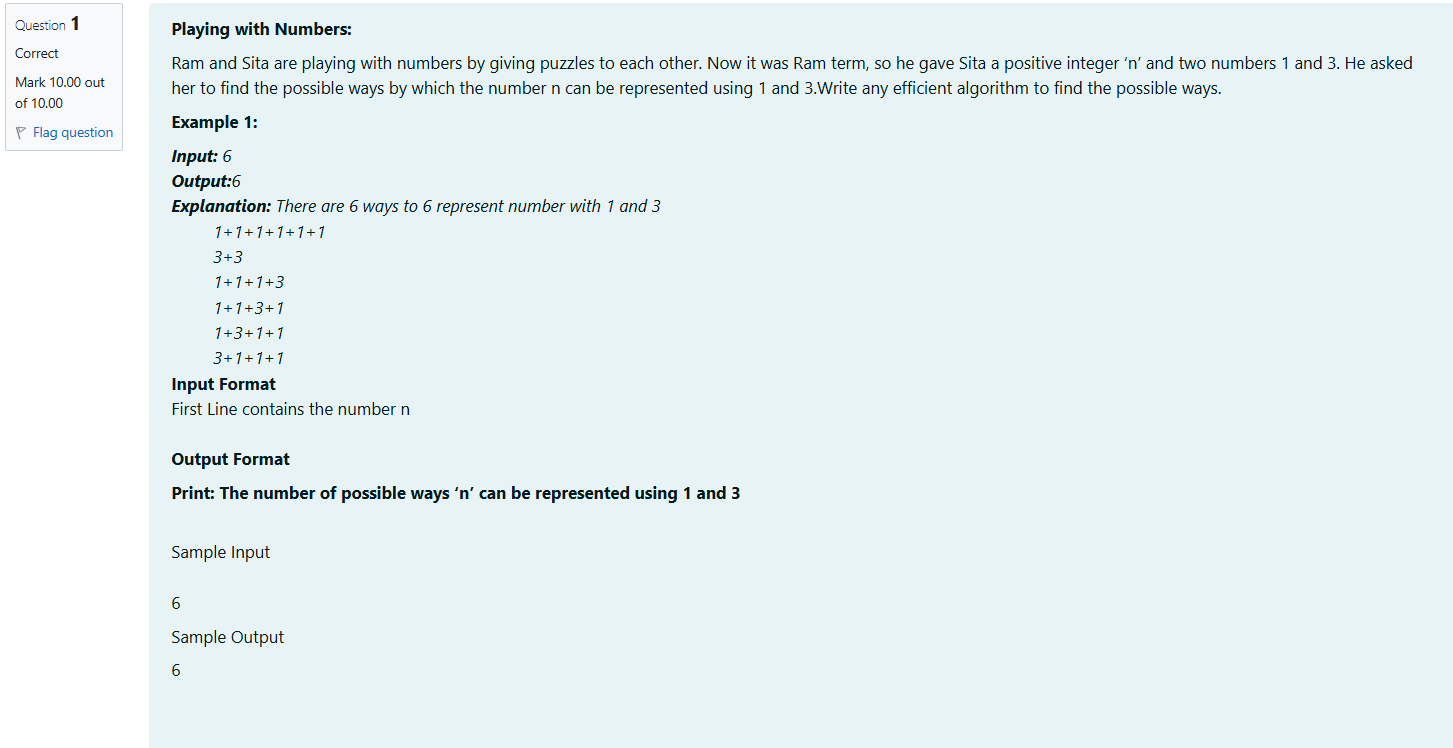
**NAME : P.R.DHANVIN**

**DEPT & SEC : CSE & ‘B’**

**ROLL NO : 230701071**

**WEEK : DYNAMIC PROGRAMMING**

**1-DP-PLAYING WITH NUMBERS**



**#include<stdio.h>**

**long long int count(int n){**

**long long int table[n+1];**

**int i;**

**for(i=0; i<n+1; i++){**

**table[i] = 0;**

**}**

**table[0] = 1;**

**for (int i = 1; i <= n; i++) {**

**table[i] += table[i - 1];**

**if (i >= 3) {**

**table[i] += table[i - 3];**

**}**

**}**

**return table[n];**

**}**

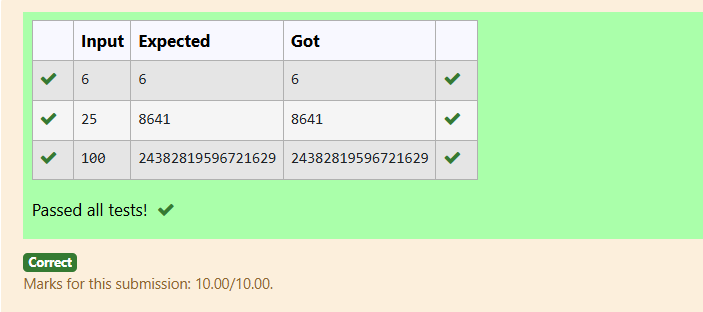
**int main(){**

**int n;**

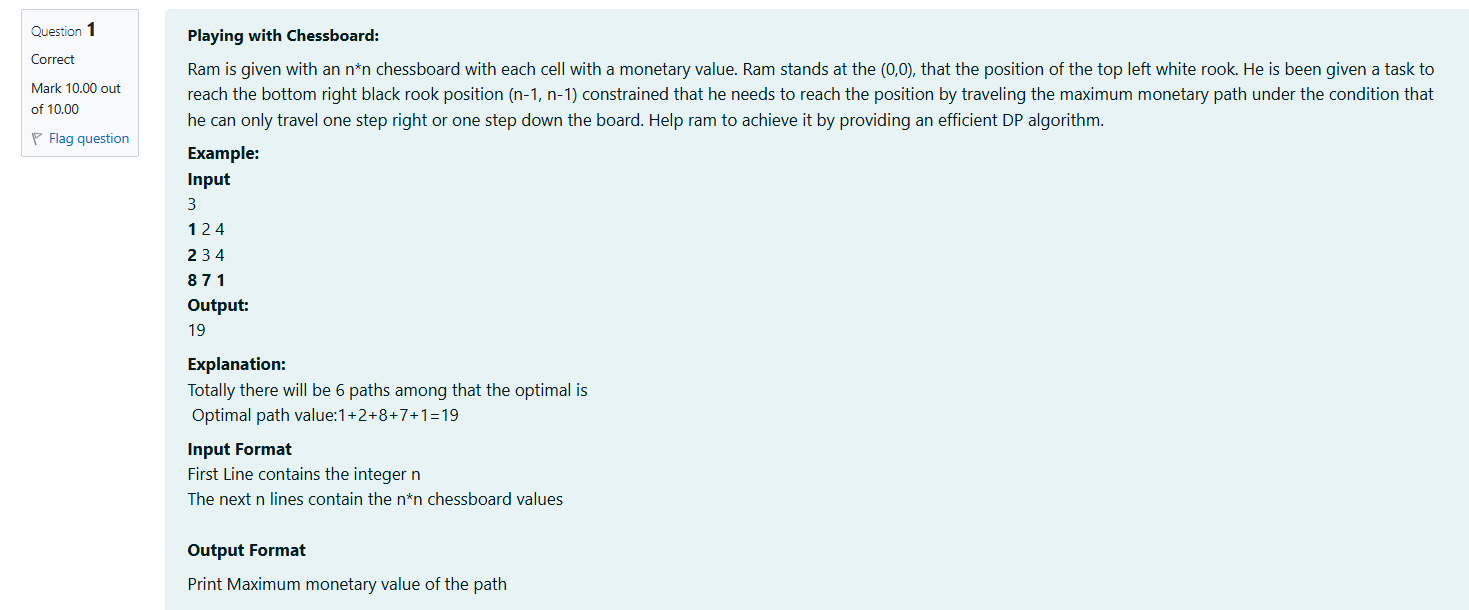
**scanf("%d", &n);**

**printf("%lld", count(n));**

**}**



**2-DP-PLAYING WITH CHESSBOARD**



**#define MAX\_N 100**

**#include<stdio.h>**

**int max(int a, int b) {**

**return (a > b) ? a : b;**

**}**

**int main() {**

**int n;**

**int board[MAX\_N][MAX\_N];**

**int dp[MAX\_N][MAX\_N];**

**scanf("%d", &n);**

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

**for (int j = 0; j < n; j++) {**

**scanf("%d", &board[i][j]);**

**}**

**}**

**dp[0][0] = board[0][0];**

**for (int i = 1; i < n; i++) {**

**dp[i][0] = dp[i-1][0] + board[i][0];**

**}**

**for (int j = 1; j < n; j++) {**

**dp[0][j] = dp[0][j-1] + board[0][j];**

**}**

**for (int i = 1; i < n; i++) {**

**for (int j = 1; j < n; j++) {**

**dp[i][j] = max(dp[i-1][j], dp[i][j-1]) + board[i][j];**

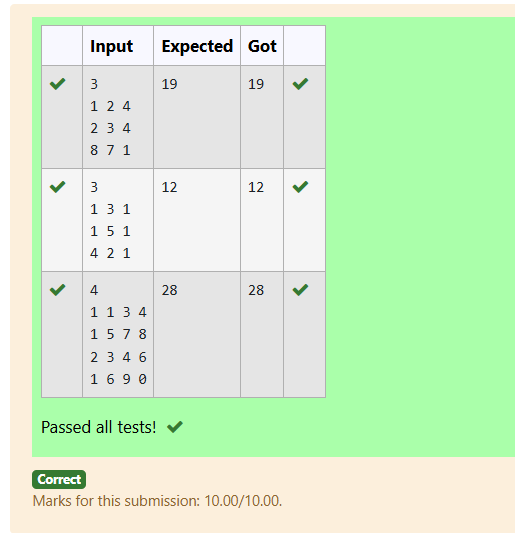
**}**

**}**

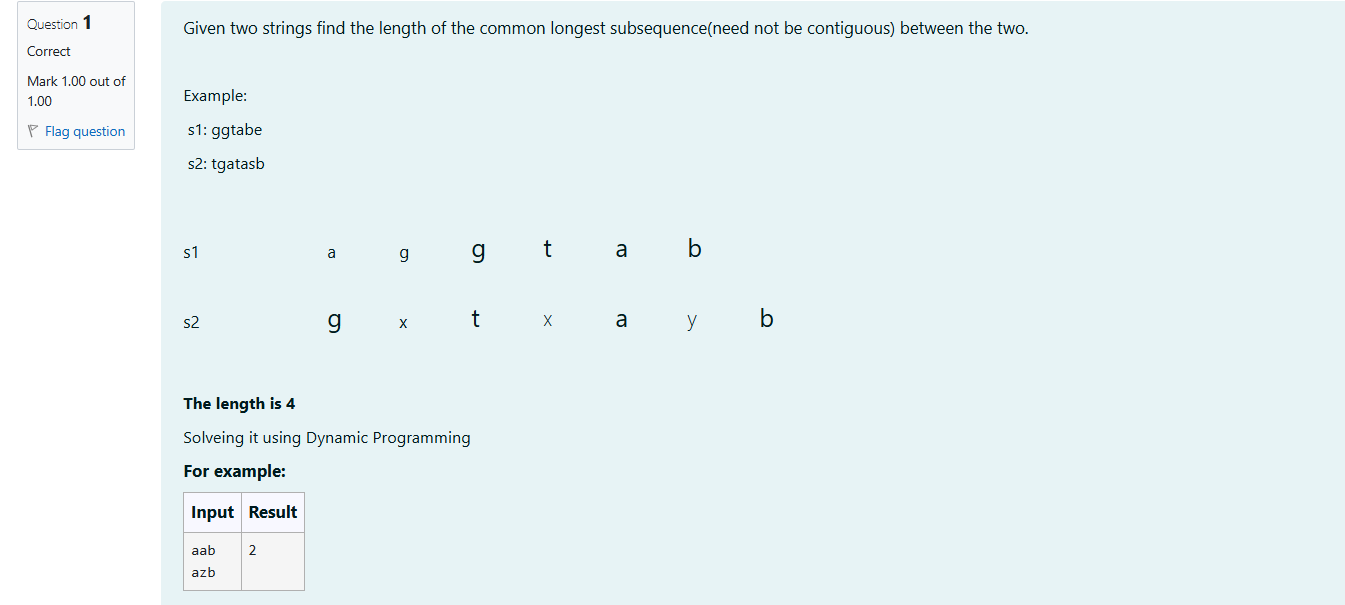
**printf("%d\n", dp[n-1][n-1]);**

**return 0;**

**}**



**3-DP-LONGEST COMMON SUBSEQUENCE**



**#include <stdio.h>**

**#include <string.h>**

**#define M 100**

**int max(int a, int b) {**

**return (a > b) ? a : b;**

**}**

**int main() {**

**char s1[M], s2[M];**

**int dp[M+1][M+1];**

**scanf("%s", s1);**

**scanf("%s", s2);**

**int n = strlen(s1);**

**int m = strlen(s2);**

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

**for (int j = 0; j <= m; j++) {**

**if (i == 0 || j == 0)**

**dp[i][j] = 0;**

**else if (s1[i-1] == s2[j-1])**

**dp[i][j] = dp[i-1][j-1] + 1;**

**else**

**dp[i][j] = max(dp[i-1][j], dp[i][j-1]);**

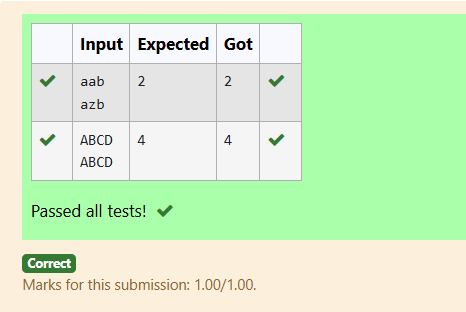
**}**

**}**

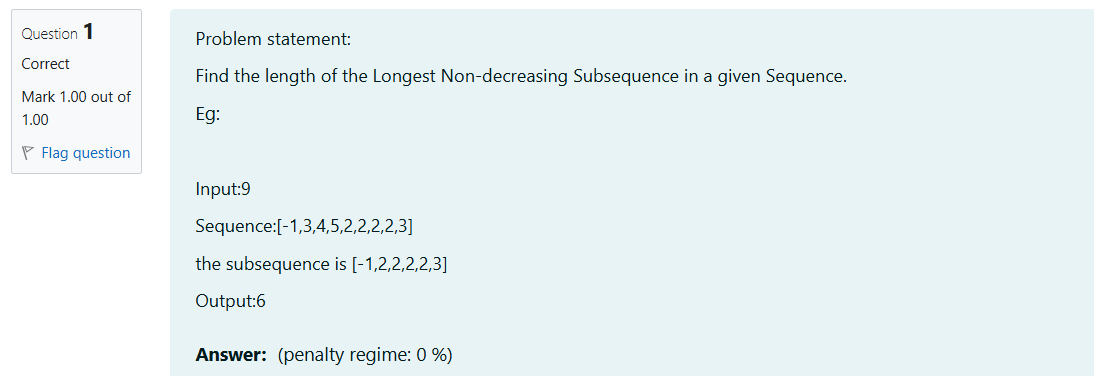
**printf("%d\n", dp[n][m]);**

**return 0;**

**}**



**4-DP-LONGEST NON-DECREASING SUBSEQUENCE**



**#include <stdio.h>**

**#define MAX 100**

**int main() {**

**int n;**

**int s[MAX];**

**int a[MAX];**

**scanf("%d",&n);**

**for (int i=0;i<n;i++) {scanf("%d",&s[i]);}**

**int m= 1;**

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

**a[i] = 1;**

**for (int j = 0; j < i; j++) {**

**if (s[j] <= s[i]) {a[i] = a[i]>a[j]+1?a[i]:a[j]+1;}**

**}**

**if (a[i] > m) {m = a[i];}**

**}**

**printf("%d\n", m);**

**return 0;**

**}**

