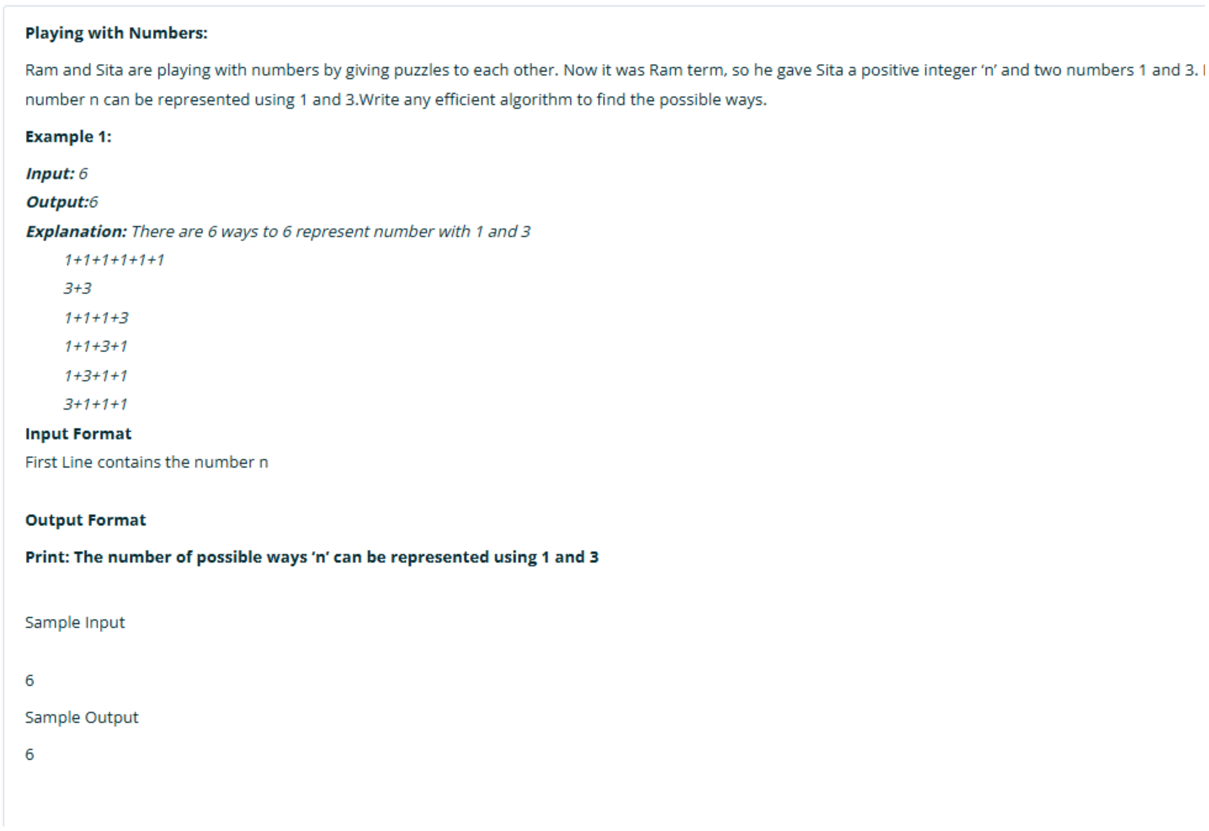
RESHMA S 2024-CSE

Dynamic programming

1-DP-Playing with Numbers



Playing with Numbers:

Ram and Sita are playing with numbers by giving puzzles to each other. Now it was Ram term, so he gave Sita a positive integer 'n' and two numbers 1 and 3. I
number n can be represented using 1 and 3.Write any efficient algorithm to find the possible ways.

Example 1:

Input: *6*
Output:*6*
Explanation: *There are 6 ways to 6 represent number with 1 and 3*

1+1+1+1+1+1
3+3
1+1+1+3
1+1+3+1
1+3+1+1
3+1+1+1

Input Format
First Line contains the number n

Output Format

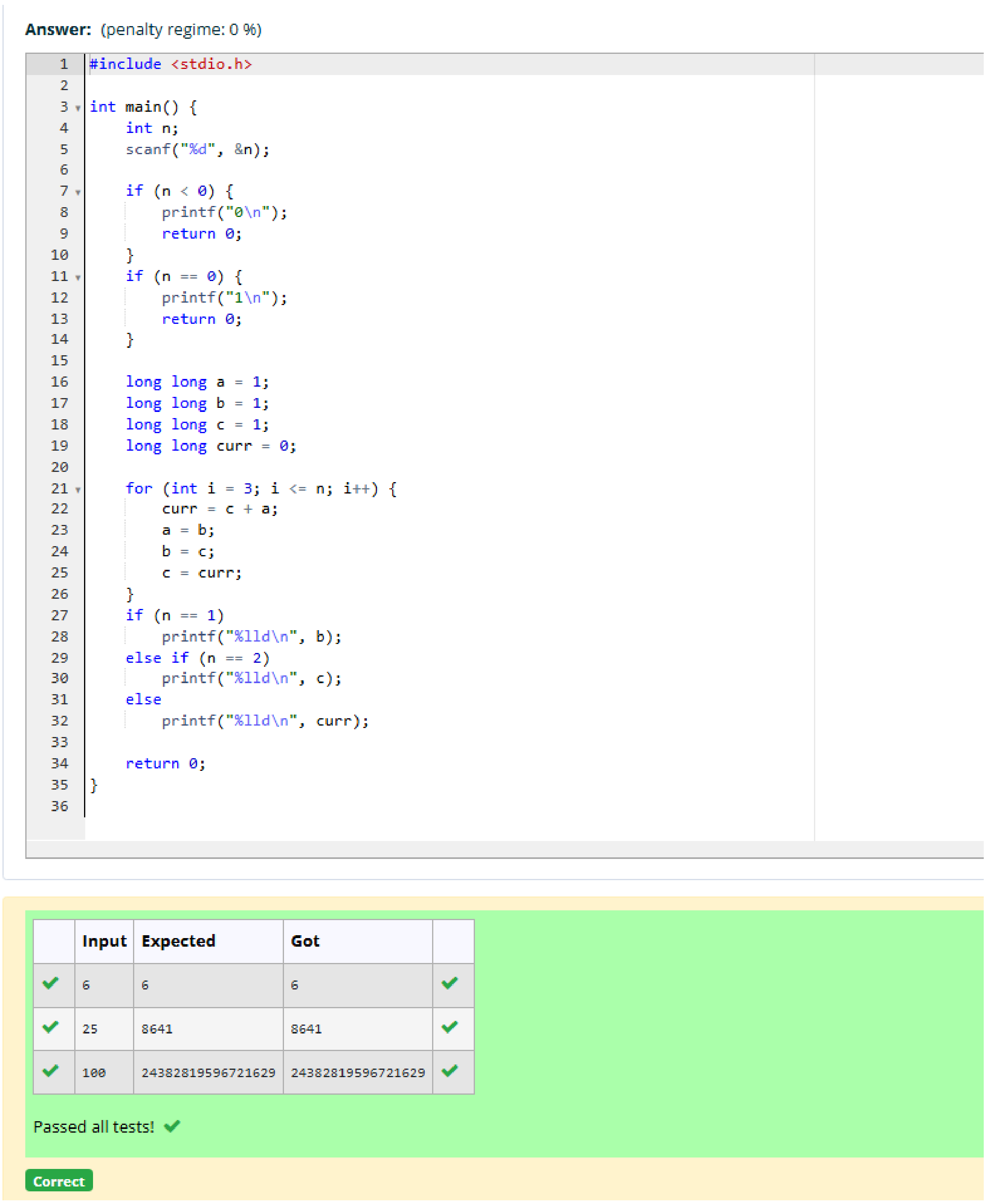
Print: The number of possible ways 'n' can be represented using 1 and 3

Sample Input

6

Sample Output

6



Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

int main() {
    int n;
    scanf("%d", &n);

    if (n < 0) {
        printf("0\n");
        return 0;
    }
    if (n == 0) {
        printf("1\n");
        return 0;
    }

    long long a = 1;
    long long b = 1;
    long long c = 1;
    long long curr = 0;

    for (int i = 3; i <= n; i++) {
        curr = c + a;
        a = b;
        b = c;
        c = curr;
    }
    if (n == 1)
        printf("%lld\n", b);
    else if (n == 2)
        printf("%lld\n", c);
    else
        printf("%lld\n", curr);

    return 0;
}

```

	Input	Expected	Got	
✓	6	6	6	✓
✓	25	8641	8641	✓
✓	100	24382819596721629	24382819596721629	✓

Passed all tests! ✓

Correct

2-DP-Playing with chessboard

Question 1 | Correct Mark 10.00 out of 10.00 Flag question

Playing with Chessboard:

Ram is given with an n*n chessboard with each cell with a monetary value. Ram stands at the (0,0), that the position of the top left white rook. He is been given a task
(n-1, n-1) constrained that he needs to reach the position by traveling the maximum monetary path under the condition that he can only travel one step right or one s
providing an efficient DP algorithm.

Example:

Input

3

1 2 4

2 3 4

8 7 1

Output:

19

Explanation:

Totally there will be 6 paths among that the optimal is

Optimal path value:1+2+8+7+1=19

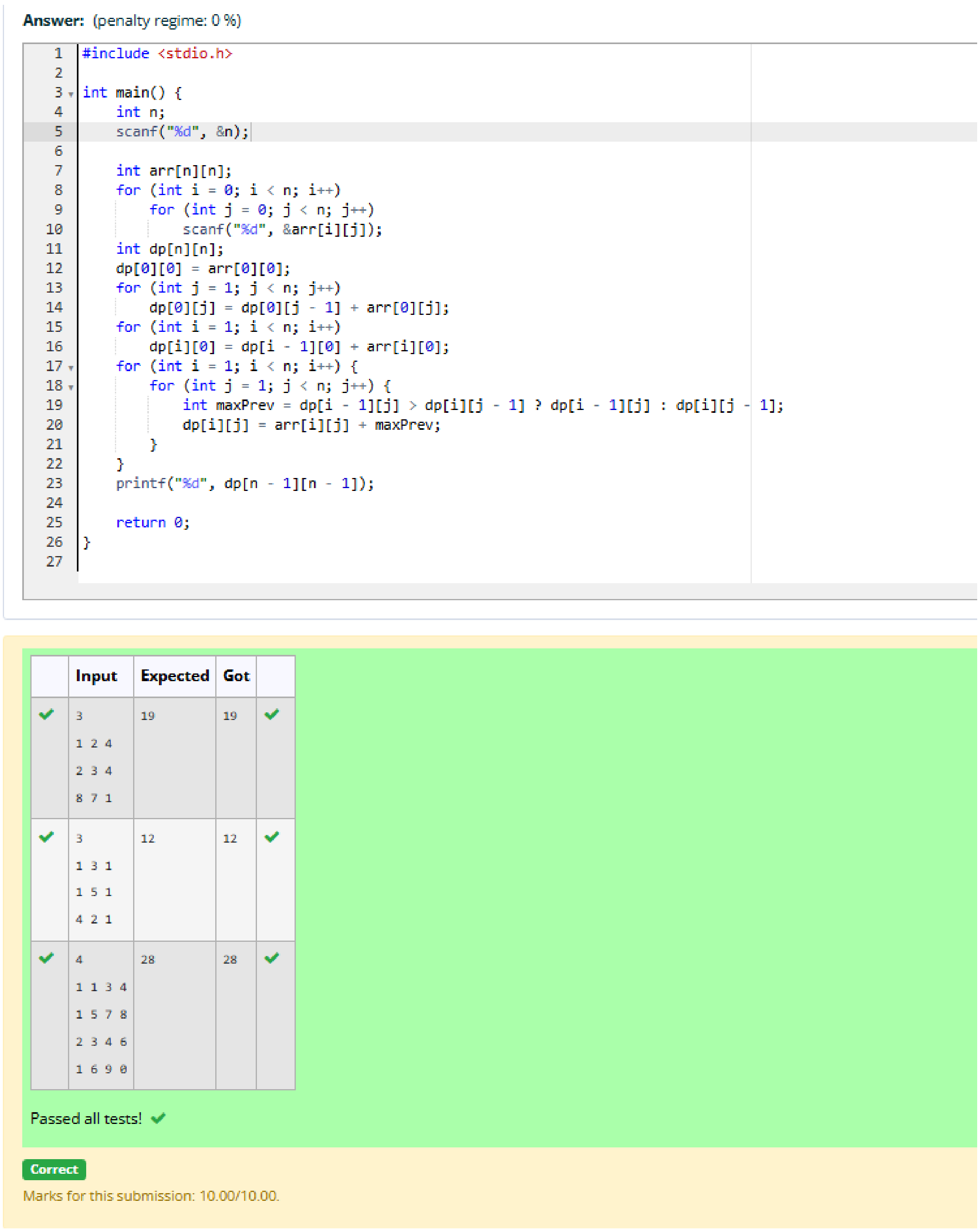
Input Format

First Line contains the integer n

The next n lines contain the n*n chessboard values

Output Format

Print Maximum monetary value of the path



Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

int main() {
    int n;
    scanf("%d", &n);

    int arr[n][n];
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            scanf("%d", &arr[i][j]);
    int dp[n][n];
    dp[0][0] = arr[0][0];
    for (int j = 1; j < n; j++)
        dp[0][j] = dp[0][j - 1] + arr[0][j];
    for (int i = 1; i < n; i++)
        dp[i][0] = dp[i - 1][0] + arr[i][0];
    for (int i = 1; i < n; i++) {
        for (int j = 1; j < n; j++) {
            int maxPrev = dp[i - 1][j] > dp[i][j - 1] ? dp[i - 1][j] : dp[i][j - 1];
            dp[i][j] = arr[i][j] + maxPrev;
        }
    }
    printf("%d", dp[n - 1][n - 1]);

    return 0;
}
```

	Input	Expected	Got	
✓	3 1 2 4 2 3 4 8 7 1	19	19	✓
✓	3 1 3 1 1 5 1 4 2 1	12	12	✓
✓	4 1 1 3 4 1 5 7 8 2 3 4 6 1 6 9 0	28	28	✓

Passed all tests! ✓

Correct

Marks for this submission: 10.00/10.00.

3-DP-Longest Common Subsequence

Given two strings find the length of the common longest subsequence(need not be contiguous) between the two.

Example:

s1: ggtabe

s2: tgatasb

s1	a	g	g	t	a	b	
s2	g	x	t	x	a	y	b

The length is 4

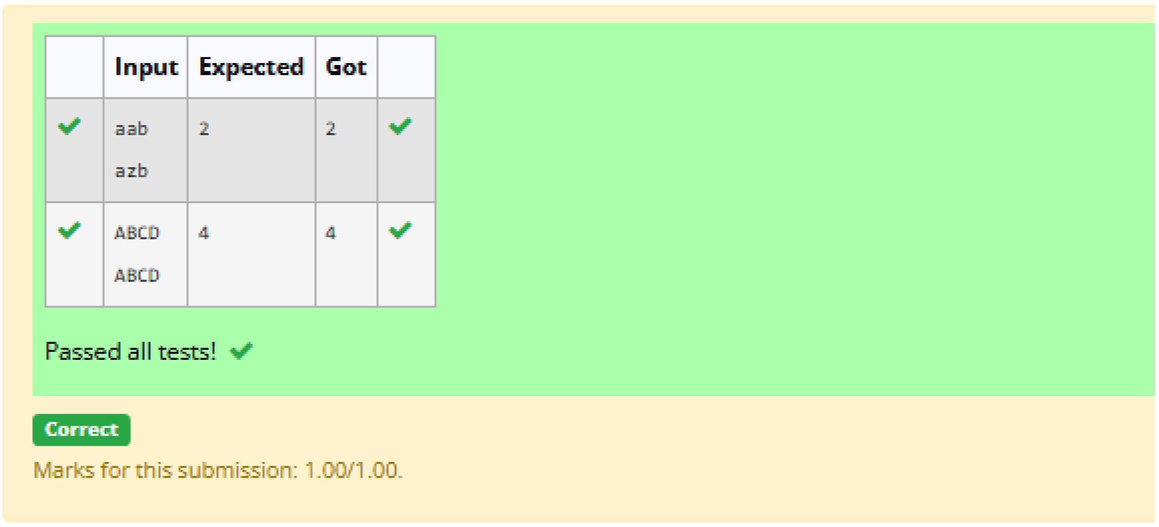
Solveing it using Dynamic Programming

For example:

Input	Result
aab azb	2

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>

int max(int a, int b) {
    return (a > b) ? a : b;
}

int main() {
    char s1[100], s2[100];
    scanf("%s %s", s1, s2);

    int n = strlen(s1);
    int m = strlen(s2);
    int dp[n + 1][m + 1];
    for (int i = 0; i <= n; i++)
        for (int j = 0; j <= m; j++)
            dp[i][j] = 0;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= m; j++) {
            if (s1[i - 1] == s2[j - 1])
                dp[i][j] = 1 + dp[i - 1][j - 1];
            else
                dp[i][j] = max(dp[i - 1][j], dp[i][j - 1]);
        }
    }

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

    return 0;
```



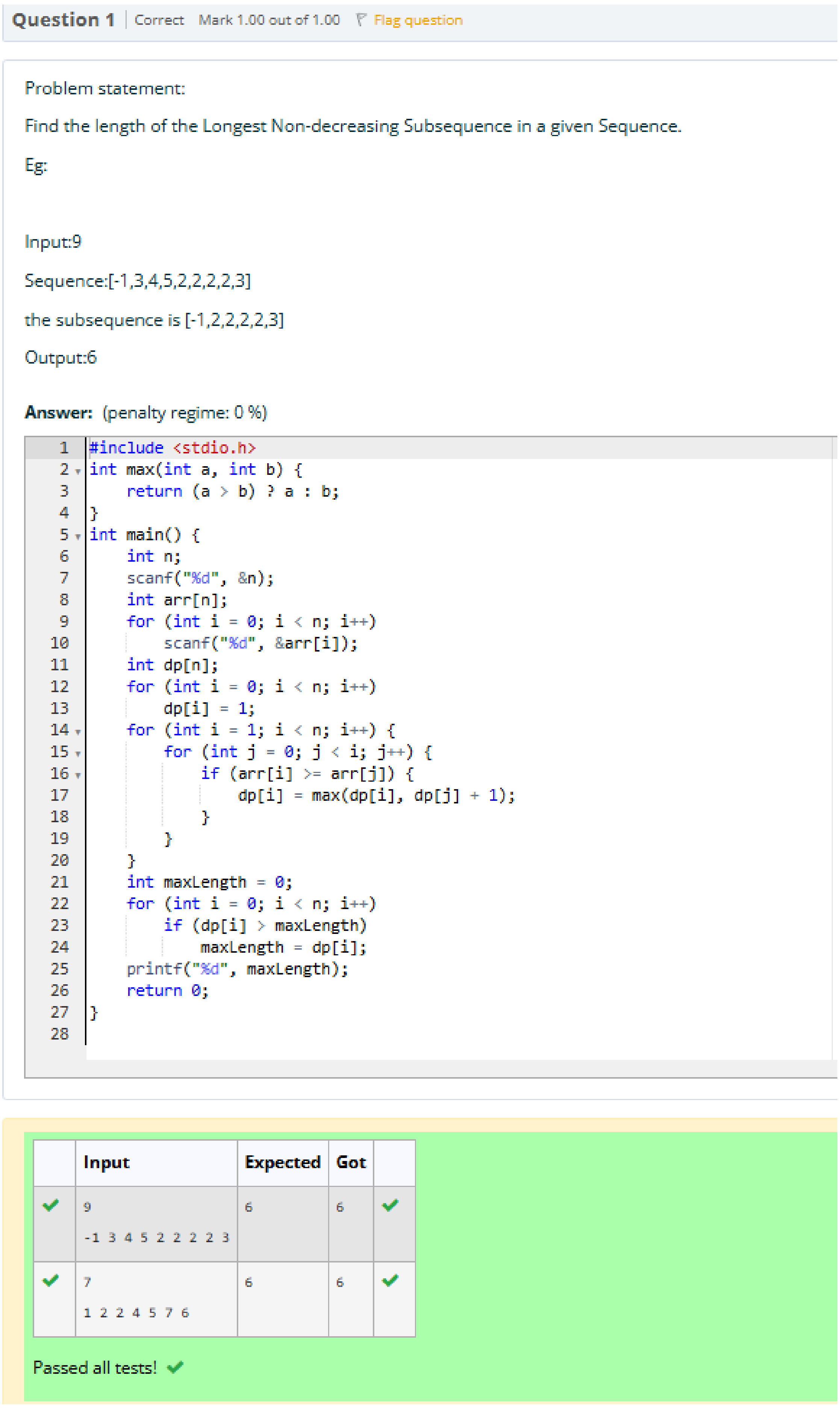
	Input	Expected	Got	
✓	aab azb	2	2	✓
✓	ABCD ABCD	4	4	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

4-DP-Longest non-decreasing Subsequence



Question 1 | Correct Mark 1.00 out of 1.00 Flag question

Problem statement:

Find the length of the Longest Non-decreasing Subsequence in a given Sequence.

Eg:

Input:9

Sequence:[-1,3,4,5,2,2,2,2,3]

the subsequence is [-1,2,2,2,2,3]

Output:6

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int max(int a, int b) {
    return (a > b) ? a : b;
}
int main() {
    int n;
    scanf("%d", &n);
    int arr[n];
    for (int i = 0; i < n; i++)
        scanf("%d", &arr[i]);
    int dp[n];
    for (int i = 0; i < n; i++)
        dp[i] = 1;
    for (int i = 1; i < n; i++) {
        for (int j = 0; j < i; j++) {
            if (arr[i] >= arr[j]) {
                dp[i] = max(dp[i], dp[j] + 1);
            }
        }
    }
    int maxLength = 0;
    for (int i = 0; i < n; i++)
        if (dp[i] > maxLength)
            maxLength = dp[i];
    printf("%d", maxLength);
    return 0;
}

```

	Input	Expected	Got	
✓	9 -1 3 4 5 2 2 2 2 3	6	6	✓
✓	7 1 2 2 4 5 7 6	6	6	✓

Passed all tests! ✓