

- MODEL EXAM AIDS,AIML,IT
- BASIC C PROGRAMMING
- Finding Time Complexity o...
- Divide and Conquer
- Greedy Algorithms
- Dynamic Programming
- 1-DP-Playing with Numbers
- 2-DP-Playing with chessboard
- 3-DP-Longest Common Subs...
- 4-DP-Longest non-decreasin...
- Competitive Programming
- 1-Finding Duplicates-O(n^2) ...
- 2-Finding Duplicates-O(n) Ti...
- 3-Print Intersection of 2 sort...
- 4-Print Intersection of 2 sort...
- 5-Pair with Difference-O(n^2...)
- 6-Pair with Difference -O(n) ...

DIVYASH H 2024-CSE

D2

Dashboard My courses

CS23331-DAA-2024-CSE / 3-DP-Longest Common Subsequence

3-DP-Longest Common Subsequence

Started on	Sunday, 19 October 2025, 7:49 PM
State	Finished
Completed on	Sunday, 19 October 2025, 7:51 PM
Time taken	2 mins 27 secs
Marks	1.00/1.00
Grade	10.00 out of 10.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

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

Solving it using Dynamic Programming

For example:

Input	Result
aab	2
azb	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 #include <string.h>
3
4 int max(int a, int b) {
5     return (a > b) ? a : b;
6 }
7
8 int main() {
9     char s1[1000], s2[1000];
10    scanf("%s", s1);
11    scanf("%s", s2);
12
13    int m = strlen(s1);
14    int n = strlen(s2);
15
16    int dp[m + 1][n + 1];
17
18    for (int i = 0; i <= m; i++) {
19        for (int j = 0; j <= n; j++) {
20            dp[i][j] = 0;
21
22        for (int i = 1; i <= m; i++) {
23            for (int j = 1; j <= n; j++) {
24                if (s1[i - 1] == s2[j - 1])
25                    dp[i][j] = 1 + dp[i - 1][j - 1];
26                else
27                    dp[i][j] = max(dp[i - 1][j], dp[i][j - 1]);
28            }
29        }
30
31    printf("%d\n", dp[m][n]);
32    return 0;
33 }
34 }
```

1
✓

Finish review

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Finish review](#)

[Back to Course](#)

Data retention summary

- > BASIC C PROGRAMMING
- > Finding Time Complexity o...
- > Divide and Conquer
- > Greedy Algorithms
- > Dynamic Programming
- 1-DP-Playing with Numbers
- 2-DP-Playing with chessboard
- 3-DP-Longest Common Subs...
- 4-DP-Longest non-decreasin...**
- > Competitive Programming
- 1-Finding Duplicates-O(n^2) ...
- 2-Finding Duplicates-O(n) Ti...
- 3-Print Intersection of 2 sort...
- 4-Print Intersection of 2 sort...
- 5-Pair with Difference-O(n^2...)
- 6-Pair with Difference -O(n) ...

**RAJALAKSHMI
ENGINEERING
COLLEGE**
DIVYASH H 2024-CSE

D2
D3

Dashboard My courses

CS23331-DAA-2024-CSE / 4-DP-Longest non-decreasing Subsequence

4-DP-Longest non-decreasing Subsequence

Started on: Sunday, 19 October 2025, 7:52 PM

State: Finished

Completed on: Sunday, 19 October 2025, 7:56 PM

Time taken: 4 mins 38 secs

Marks: 1.00/1.00

Grade: **10.00** out of 10.00 (100%)

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

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

Output:6

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2
3 int max(int a, int b) {
4     return (a > b) ? a : b;
5 }
6
7 int main() {
8     int n;
9     scanf("%d", &n);
10    int arr[n], dp[n];
11
12    for (int i = 0; i < n; i++) {
13        scanf("%d", &arr[i]);
14        dp[i] = 1;
15    }
16
17    for (int i = 1; i < n; i++) {
18        for (int j = 0; j < i; j++) {
19            if (arr[i] >= arr[j]) {
20                dp[i] = max(dp[i], dp[j] + 1);
21            }
22        }
23    }
24
25    int result = 0;
26    for (int i = 0; i < n; i++) {
27        result = max(result, dp[i]);
28    }
29
30    printf("%d\n", result);
31    return 0;
32 }
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Quiz navigation

1 ✓

[Finish review](#)

[Finish review](#)

[Back to Course](#)

Data retention summary

x

O MODEL EXAM CSE Batch 1

O MODEL EXAM AIDS,AIML,IT

> BASIC C PROGRAMMING

> Finding Time Complexity o...

> Divide and Conquer

> Greedy Algorithms

> Dynamic Programming

● 1-DP-Playing with Numbers

● 2-DP-Playing with chessboard

● 3-DP-Longest Common Subs...

● 4-DP-Longest non-decreasin...

> Competitive Programming

● 1-Finding Duplicates-O(n^2) ...

● 2-Finding Duplicates-O(n) Ti...

● 3-Print Intersection of 2 sort...

● 4-Print Intersection of 2 sort...

● 5-Pair with Difference-O(n^2...



DIVYASH H 2024-CSE

D2



Dashboard My courses



CS23331-DAA-2024-CSE / 2-DP-Playing with chessboard

2-DP-Playing with chessboard

Started on Monday, 3 November 2025, 8:45 AM

State Finished

Completed on Monday, 3 November 2025, 8:51 AM

Time taken 5 mins 21 secs

Grade 10.00 out of 10.00 (100%)

Question 1 | Correct Mark 10.00 out of 10.00 Flag question

Playing with Chessboard:

Ram is given with an $n \times n$ chessboard with each cell with a monetary value. Ram stands at the (0,0), that is the position of the top left white rook. He is given a task to reach the bottom right black rook position ($n-1, n-1$) 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 step down the board. Help ram to achieve it by 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 $n \times n$ chessboard values

Output Format

Print Maximum monetary value of the path

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int max(int a, int b) {
4     return (a > b) ? a : b;
5 }
6
7 int main() {
8     int n;
9     scanf("%d", &n);
10
11    int board[n][n];
12
13    for (int i = 0; i < n; i++) {
14        for (int j = 0; j < n; j++) {
15            scanf("%d", &board[i][j]);
16
17        for (int i = 0; i < n; i++) {
18            for (int j = 0; j < n; j++) {
19                if (i > 0 && j > 0)
20                    board[i][j] += max(board[i - 1][j], board[i][j - 1]);
21                else if (i > 0)
22                    board[i][j] += board[i - 1][j];
23                else if (j > 0)
24                    board[i][j] += board[i][j - 1];
25            }
26        }
27
28        printf("%d\n", board[n - 1][n - 1]);
29
30    return 0;
31 }
32

```

Quiz navigation

1

✓

Finish review

	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.

[Finish review](#)

[Back to Course](#)

Data retention summary

x

- General
 - MODEL EXAM CSE Batch 1
 - MODEL EXAM AIDS,AIML,IT
- BASIC C PROGRAMMING
- Finding Time Complexity o...
- Divide and Conquer
- Greedy Algorithms
- Dynamic Programming
 - 1-DP-Playing with Numbers**
 - 2-DP-Playing with chessboard
 - 3-DP-Longest Common Subs...
 - 4-DP-Longest non-decreasin...
- Competitive Programming
 - 1-Finding Duplicates-O(n^2) ...
 - 2-Finding Duplicates-O(n) Ti...
 - 3-Print Intersection of 2 sort...
 - 4-Print Intersection of 2 sort...

RAJALAKSHMI
ENGINEERING
COLLEGE

DIVYASH H 2024-CSE D2

Dashboard My courses

CS23331-DAA-2024-CSE / 1-DP-Playing with Numbers

1-DP-Playing with Numbers

Started on	Wednesday, 29 October 2025, 10:31 AM
State	Finished
Completed on	Wednesday, 29 October 2025, 10:42 AM
Time taken	11 mins 7 secs
Grade	10.00 out of 10.00 (100%)

Question 1 | Correct Mark 10.00 out of 10.00 [Flag question](#)

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. He asked her to find the possible ways by which the 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 represent number with 1 and 3

$$\begin{aligned} &1+1+1+1+1+1 \\ &3+3 \\ &1+1+1+3 \\ &1+1+3+1 \\ &1+3+1+1 \\ &3+1+1+1 \end{aligned}$$

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 %)

```

1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     if(n==0){
7         printf("0\n");
8         return 0;
9     }
10    if(n==1){
11        printf("1\n");
12        return 0;
13    }
14    if(n==2){
15        printf("2\n");
16        return 0;
17    }
18    unsigned long long f0 = 1;
19    unsigned long long f1 = 1;
20    unsigned long long f2 = 1;
21    unsigned long long fn = 0;
22    for(int i = 3; i<=n;i++){
23        fn=f2+f0;
24        f0=f1;
25        f1=f2;
26        f2=fn;
27    }
28    printf("%llu\n",fn);
29    return 0;
30 }
```

Quiz navigation

1 ✓

Finish review

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

Passed all tests! ✓

Correct

Marks for this submission: 10.00/10.00.

[Finish review](#)

[Back to Course](#)

Data retention summary