



## DJ Codestars Annual Report 2020-21

### Core Committee for Annual Year 2020-21

Position	Position Holders
Chairpersons	Smit Malkan and Ankit Gupta
Technical Heads	Rayyan Merchant, Nilay Bhatia, Jash Jain, Preet Soni and Riddham Gadia
Marketing and Publicity Heads	Deep Nanda and Kevin Pattni
Editorial Head	Bhagyashree Wagh

The team organised 8 lectures, 2 contests and 1 additional event in the period of September 2020 to November 2020, under the extremely helpful guidance of our faculty mentors:

- Dr. Vinaya Sawant
- Prof. Aruna Gawde
- Prof. Neha Katre
- Prof. Pranit Bari

In addition to the core and faculties, a team of 12 hardworking co-committee members has been handling the Instagram account, organising the lectures, collecting resources for teaching, helping decide the problems for the contests and much more.



## Events

### 1. Introduction to Competitive Programming

**Date and Time:** 24th September 2020, 4:30 pm

This open to all session was conducted primarily by chairpersons Ankit Gupta and Smit Malkan, where they spoke about the importance of competitive programming from a placements perspective, as well as a skill-building perspective. Second and Third-year students from almost all branches of DJ Sanghvi attended the session in large numbers, and showed a heightened interest in competitive programming, as evident from the high attendance in future lectures.

## Lectures

**A:** The following lectures were conducted for SEs, TEs and BEs.

### 1. Lecture #1 on Time complexities

**Date and Time:** 29th September, 4:30pm.

Being the first lecture, it focused on more of the basics in CP. It focused on how to understand the time complexity requirements of the problem and showed live examples of problems with their solutions. It was conducted by <names>. Around <number> attended this lecture from <branches>.

**Link:** <https://web.microsoftstream.com/video/c7258fa8-2040-4e86-8ac4-e879b2aae3dd>

Presenting...

```
int a = 0, i = N;
while (i > 0) {
    a += i;
    i /= 2;
}
```

Options:

1. O(N)
2. O(Sqrt(N))
3. O(N / 2)
4. O(log N)

Handwritten notes:  $\frac{N}{2^k} = 1$ ,  $\frac{N}{2} = 1$ ,  $\dots \frac{N}{2} = 1$

0:56:18 / 2:17:18



## 2. Lecture #2 on Inputs, outputs, sorting and binary search.

**Date and Time:** 2nd October, 2:30 pm.

Firstly the lecture started with the basics of how to input the user data and how to output the computed data back to the user. And then it focussed on two of the fundamental operations which were sorting and searching in an efficient way. Both the algorithm were traced and their time complexity was derived respectively. This lecture was conducted by Riddham Gadia, Smit Malkan, Nilay Bhatia, and Jash Jain. Around 35 students attended that lecture.

**Link:** <https://web.microsoftstream.com/video/00fea343-8029-477f-97d6-b2fc42d58b0d>

```
11 for(int i = 0; i < n; ++ i)
12 {
13     scanf("%d", &a[i]);
14 }
15 int su = 0;
16 for(int i = 0; i < n; ++ i)
17 {
18     su += a[i];
19 }
20 printf("Sum = %d\n", su);
21 printf("Average = %0.15f", (double) su / n);
22 sort(a, a + n);
23 reverse(a, a + n);
24 for(int i = 0; i < n; ++ i)
25     printf("%d ", a[i]);
26     printf("\n");
27
28 int disp[2][4] = {
29     {10, 11, 12, 13}
```

## 3. Lecture #3 on Stacks and queues

**Date and Time:** 6th October, 4:30pm.

The lecture started by showing a runthrough of basic stack operations like .push .pop .empty and .top and also a demo on how to use the debugger in VSCode to show the working of the code in a step by step form. Problems on stacks and queues were shown and solved using the basic stack and queue operations taught earlier in the lecture. The lecture was taught by Smit Malkan, Jash Jain and Preet Soni and around 45 students attended that lecture.

**Link:** <https://web.microsoftstream.com/video/09cfb669-0505-4e81-ab44-5ba07e041fea>

```
31 freopen("input.txt", "r", stdin);
32 freopen("output.txt", "w", stdout);
33 freopen("error.txt", "w", stderr);
34 #endif
35
36 int num;
37 scanf("%d",&num);
38
39 stack<int> s;
40
41 for(int i=0;i<num;i++)
42 {
43     int temp;
44     scanf("%d", &temp);
45     while(!s.empty() && s.top() < temp){
46         printf("%d -> %d\n", s.top(), temp);
47         s.pop();
48     }
49     s.push(temp);
50
51 while(!s.empty()){
52     printf("%d -> -1\n", s.top());
53     s.pop();
54 }
55 //cerr<<"time taken : "<<(float)clock()/CLOCKS_PER_SEC<<" secs"<<endl
```

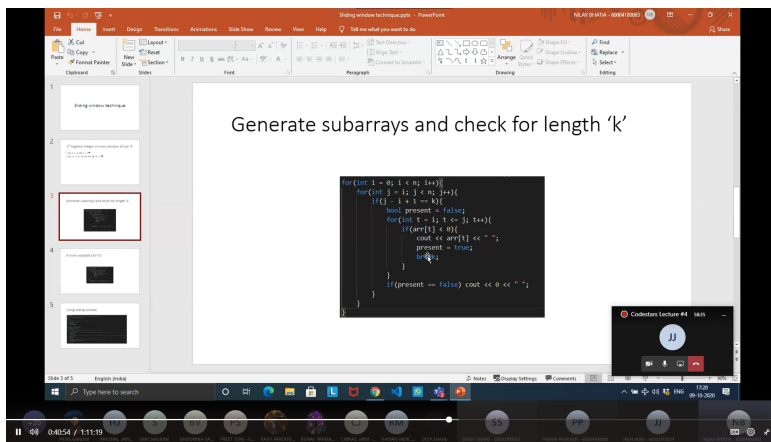


## 4. Lecture #4 on Queues and binary search.

**Date and Time:** 9th October, 4:30pm.

This lecture focussed on Binary Search, specifically using Binary search to search for the square root of a number. Students also learnt about the importance of precision using this program. The previous stack building problem was discussed. We also discussed the 'First negative integer in every window of size k'. This lecture was conducted by Jash Jain and Rayyan Merchant. Around 40 students attended the lecture.

**Link:** <https://web.microsoftstream.com/video/0300de24-fafb-4ec8-9f84-5eb27fd2c868>

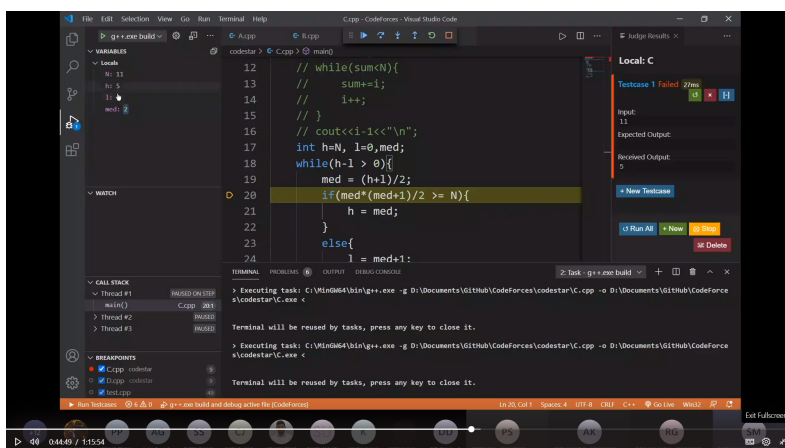


## 5. Lecture #5 on Doubts session.

**Date and Time:** 13th October, 4:30pm.

This lecture was conducted to resolve all the doubts of the students based on all the previous lectures that were held. Also the problems from the contest held on 11th October were discussed. Seniors also presented their approach to these problems. This lecture was conducted by Riddham Gadia and Smit Malkan. Around 30 students had attended the doubts session.

**Link:** <https://web.microsoftstream.com/video/887fbef4-acf9-44d1-80bb-66576595c5c9>





## 6. Lecture #6 on Hashmaps and fast exponents

**Date and Time:** 23rd October, 4:30pm.

This lecture focussed on introducing the data structure HashMaps. It began with discussing the basic concept of Hashing and then moved on to the data structure explaining time complexities of its different operations. Then a method to find fast exponents was discussed. This lecture was conducted by Riddham Gadia, Smit Malkan, Nilay Bhatia and Preet Soni. Around 35 students attended that lecture.

**Link:** <https://web.microsoftstream.com/video/78cd101d-94eb-429b-953b-81f0c7d7cd9a>

```
bool hasArrayTwoCandidates(int arr[], int n, int x) {  
    // code here  
    map<int, bool> m;  
    for(int i = 0; i < n; i++){  
        int curr = arr[i];  
        if(m[x - curr] == true){  
            return true;  
        }  
        m[curr] = true;  
    }  
    return false;  
}
```

$-2 +$   
 $-3 + 2 = -1$   
 $-2 - (-3) = 1$   
{0, -1, 2, -3, 1}, sum = -2  
M:  
0:T  
-1:T  
2:T

## 7. Lecture #7 on Recursion

**Date and Time:** 27th October, 4:30pm.

This lecture focussed on a bit more advanced Recursion. It started with a simple problem of Fibonacci to explain the basic concept then we moved to reversing a Linked List using Recursion. This lecture also focussed on Merge Sort which uses Recursion for sorting and in the end the homework problem was discussed. This lecture was conducted by Jash Jain, Rayyan Merchant and Riddham Gadia. Around 20 students attended this lecture

**Link:** <https://web.microsoftstream.com/video/93bf7b99-ba3d-46cb-9f4a-18683be705ca>

```
1 // reverse a linked list  
2 // recursive  
3 // linked list node  
4 // linked list node  
5 // linked list node  
6 // linked list node  
7 // linked list node  
8 // linked list node  
9 // linked list node  
10 // linked list node  
11 // linked list node  
12 // linked list node  
13 // linked list node  
14 // linked list node  
15 // linked list node  
16 // linked list node  
17 // linked list node  
18 // linked list node  
19 // linked list node  
20 // linked list node  
21 // linked list node  
22 // linked list node  
23 // linked list node  
24 // linked list node  
25 // linked list node  
26 // linked list node  
27 // linked list node  
28 // linked list node  
29 // linked list node  
30 // linked list node  
31 // linked list node  
32 // linked list node  
33 // linked list node  
34 // linked list node  
35 // linked list node  
36 // linked list node  
37 // linked list node  
38 // linked list node  
39 // linked list node  
40 // linked list node  
41 // linked list node  
42 // linked list node  
43 // linked list node  
44 // linked list node  
45 // linked list node  
46 // linked list node  
47 // linked list node  
48 // linked list node  
49 // linked list node  
50 // linked list node  
51 // linked list node  
52 // linked list node  
53 // linked list node  
54 // linked list node  
55 // linked list node  
56 // linked list node  
57 // linked list node  
58 // linked list node  
59 // linked list node  
60 // linked list node  
61 // linked list node  
62 // linked list node  
63 // linked list node  
64 // linked list node  
65 // linked list node  
66 // linked list node  
67 // linked list node  
68 // linked list node  
69 // linked list node  
70 // linked list node  
71 // linked list node  
72 // linked list node  
73 // linked list node  
74 // linked list node  
75 // linked list node  
76 // linked list node  
77 // linked list node  
78 // linked list node  
79 // linked list node  
80 // linked list node  
81 // linked list node  
82 // linked list node  
83 // linked list node  
84 // linked list node  
85 // linked list node  
86 // linked list node  
87 // linked list node  
88 // linked list node  
89 // linked list node  
90 // linked list node  
91 // linked list node  
92 // linked list node  
93 // linked list node  
94 // linked list node  
95 // linked list node  
96 // linked list node  
97 // linked list node  
98 // linked list node  
99 // linked list node  
100 // linked list node
```



## 8. Lecture #8 on Dynamic programming

**Date and Time:** 30th October, 4:30pm.

This lecture focused on the basics of Dynamic programming. The Fibonacci Series, staircase problem and finding the total number of paths between two points in graph using specific paths was discussed. This lecture was conducted by Riddham Gadia, Nilay Bhatia and Smit Malkan. Around 25 students attended the lecture.

**Link:** <https://web.microsoftstream.com/video/06206830-4737-4d61-8918-94c52dc32d72>

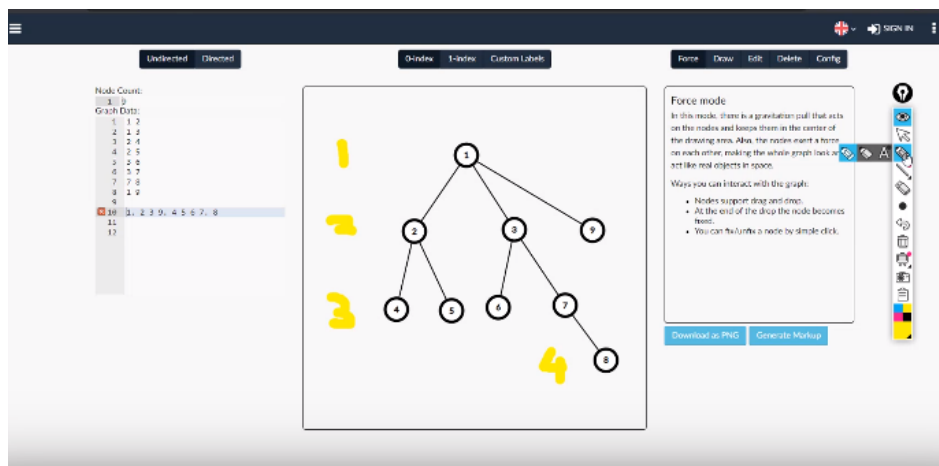
```
class fibonacci {
    static int fib(int n)
    {
        int f[] = new int[n+2];
        int i;
        // 0th and 1st number of the series are 0 and 1//
        f[0] = 0;
        f[1] = 1;
        for (i = 2; i <= n; i++)
        {
            f[i] = f[i-1] + f[i-2];
        }
        return f[n];
    }
    public static void main (String args[])
    {
        int n = 9;
        System.out.println(fib(n));
    }
}
```

## 9. Lecture #9 on Trees

**Date and Time:** 28th January, 03:00 PM

This lecture focused on the basics of Tree Data Structure. How to think Recursively in Trees problem was the main focus of the lecture. It discussed various transversal in trees like inorder, preorder, postorder, and level-order transversal. Topics such as BST, AVL trees were also touched. This lecture was conducted by Nilay Bhatia and Smit Malkan. Around 25 students attended the lecture.

**LINK :** <https://web.microsoftstream.com/video/b21eddf4-73ff-4abd-9995-4eac3fa676aa>





## 10. Lecture #10 on Graphs (#1)

Date and Time : 9th February, 03:00 PM

LINK : <https://web.microsoftstream.com/video/95aec606-0445-40a7-9bbc-0ec298b252f2>

The screenshot shows a LeetCode problem titled "743. Network Delay Time". The problem description states: "You are given a network of n nodes, labeled from 1 to n. You are also given times, a list of travel times as directed edges times[i] = (u<sub>i</sub>, v<sub>i</sub>, w<sub>i</sub>), where u<sub>i</sub> is the source node, v<sub>i</sub> is the target node, and w<sub>i</sub> is the time it takes for a signal to travel from source to target. We will send a signal from a given node k. Return the time it takes for all the n nodes to receive the signal. If it is impossible for all the n nodes to receive the signal, return -1."

An example graph is shown with nodes 1, 2, and 3. Node 1 is connected to node 2 with a weight of 1, and node 2 is connected to node 3 with a weight of 1. Node 1 is also connected to node 3 with a weight of 1.

The code solution in C++ is as follows:

```

32 for(int i=0;i<(int)dist.size();i++){
33     cout<<dist[i]<<" ";
34     if(dist[i] == max)
35         max = dist[i];
36 }
37 if(max == INT_MAX)
38     return -1;
39 else
40     return max;

```

The input is: `[[1,2,1],[1,3,1],[2,3,1]],3,4,1]]`  
 The output is: `2`

## 11. Lecture #12 on Dynamic Programming (#2)

Date and Time : 16th February, 03:00 PM

This lecture covered the problems and concepts related to Dynamic Programming including the knapsack problem. The lecture was taken by Ankit Gupta and Riddham Gadia and around 30 people attended this lecture.

LINK : <https://web.microsoftstream.com/video/6984002f-850b-4f06-87cf-021938765b20>

The screenshot shows a Microsoft Word document with handwritten notes in red ink. The text reads: "Your task is to count the number of ways to construct sum n by throwing a dice one or more times. Each throw produces an outcome between 1 and 6. For example, if n=3, there are 4 ways: 1+1+1, 1+2, 2+1, 3".

The recurrence relation is written as:  $f(n) = f(n-1) + f(n-2) + f(n-3) + \dots + f(n-6)$

The base case is written as:  $f(n) = \sum_{i=1}^6 f(n-i)$  with  $n \geq 0$  and  $f(n=0) = 1$ .

Handwritten calculations for n=3 are shown:  $3-4 = -1$ ,  $3-3 = 0$ ,  $3-2 = 1$ ,  $3-1 = 2$ .



## 12. Tech Interview Questions : Trees

Date and Time : 5th February, 03:00 PM

This lecture focussed on problems related to the data structure Tree, such as Path sum problem, Diameter of a binary tree and Lowest common ancestor. This lecture was taught by Ankit, Preet and Riddham. Around 30 students attended this lecture

LINK : <https://web.microsoftstream.com/video/0434bc27-39ee-414b-bb36-9e0746e5d242>

## 13. Tech Interview Questions: Graphs

Date and Time : 11th February, 03:00 PM

This lecture was focused on problems related to Graph, Questions were focused on the application of BFS and DFS traversal in a Graph and using BFS as a way to find the shortest path between the src and destination. This lecture was taught by Ankit, Preet, and Riddham. Around 30 students attended this lecture

LINK : <https://web.microsoftstream.com/video/4154af99-728d-4353-bbd1-6842d8a52e2a>





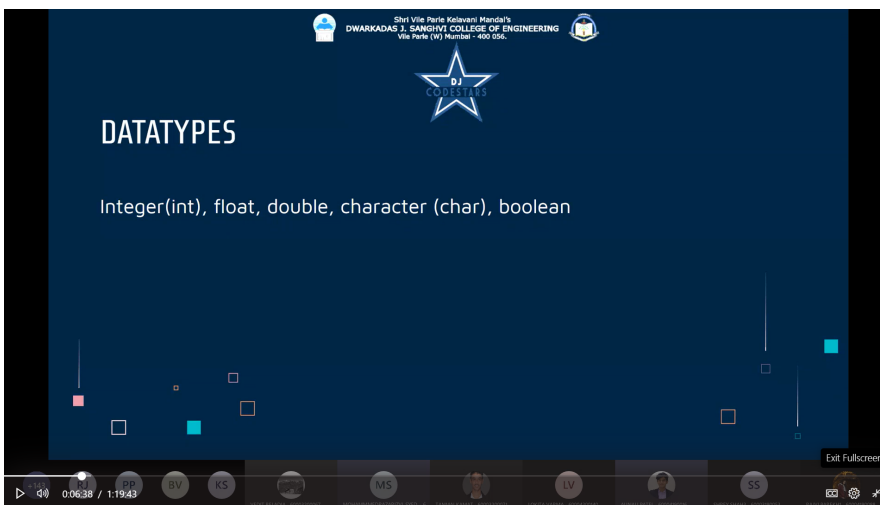
**B:** Following lectures were conducted for FEs

### Lecture #1 (Introduction to programming, variables, constants, I/O)

Date and Time: 19th February 4:00pm

This was the first lecture for FEs. We covered the basics of programming and introduced the different computer languages. Taught declaration, initialisation, use of variables, taking input and displaying output and the knowledge of basic operators. Around 200 FEs attended this lecture. This lecture was taken by Rajvi, Aunali and Parth.

**LINK:** <https://web.microsoftstream.com/video/0ab9893f-b5a3-4b98-82ba-f0b216371eff>

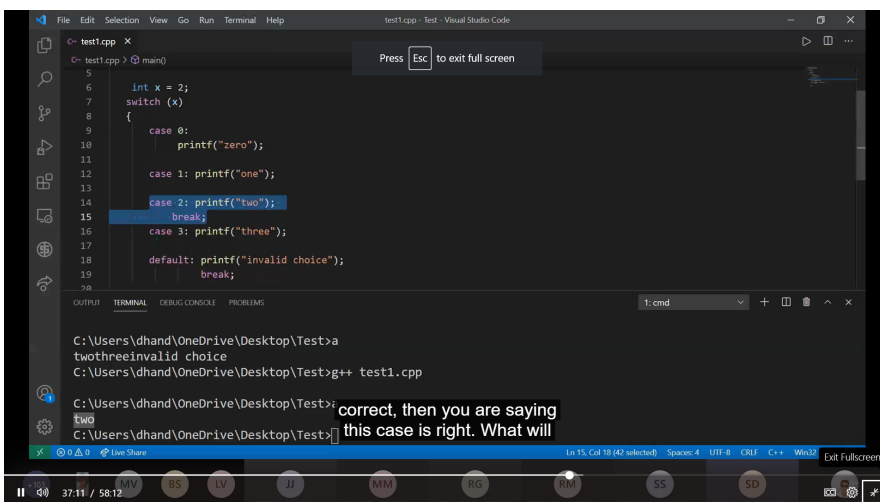


### Lecture #2 (operators, if else statements, switch case)

Date and Time: 26th February 4:00pm

In this lecture we continued with the basics of coding and concept of conditional statements- if and switch was introduced. Logical operators and ternary operator was also taught and homework problems were also discussed. This lecture was taught by Shrey and Sagar, around 120 students attended the lecture.

**LINK :** <https://web.microsoftstream.com/video/aae984f1-c64b-4f06-828e-152e3cc2e1de>





### Lecture #3: For loops, shorthand operators, char and ASCII values

Date and Time: 22th March, 4:00 pm

In this lecture, the problems given as homework in lecture 2 were discussed and solved. Problems based on basic loops and shorthand operators and ASCII values were covered. Some implementation details were also taught. This lecture was taken by Rajvi and Parth around 35 students attended this lecture.

LINK: <https://web.microsoftstream.com/video/47876cd7-4cfa-401f-8e77-6fbc5ae1d043>

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     // string s; cin>>s;
6     // int sum = 0;
7     // for (char c: s) {
8     //     sum = sum + c;
9     // }
10    char c = 'I';
11    return 0;
12 }
```

Custom Input: 12345

Suppose I put here 8 and I want to

### Lecture #4: Homework discussion

Date and Time: 4th February, 4:00 pm

In this lecture, the problems given as homework in lecture 3 were discussed and solved. Problems based on basic loops and strings were covered. Some implementation details were also taught. This lecture was taken by Parth and Sagar.

LINK: <https://web.microsoftstream.com/video/51da68f9-836e-4b71-aa19-59d1763b05b1>

**Body Mass Index** | Problem Code: BMI

Read problems statements in [Hindi](#), [Mandarin Chinese](#), [Vietnamese](#), and [Bengali](#) as well.

You are given the height  $H$  (in metres) and mass  $M$  (in kilograms) of Chef. The Body Mass Index (BMI) of a person is computed as  $\frac{M}{H^2}$ .

Report the category into which Chef falls, based on his BMI:

- Category 1: Underweight if  $BMI \leq 18$
- Category 2: Normal weight if  $BMI \in \{19, 20, \dots, 24\}$
- Category 3: Overweight if  $BMI \in \{25, 26, \dots, 29\}$
- Category 4: Obesity if  $BMI \geq 30$

**Input:**

- The first line of input will contain an integer,  $T$ , which denotes the number of testcases. Then the testcases follow.
- Each testcase contains a single line of input, with two space separated integers,  $M, H$ , which denote the mass and height of Chef respectively.

**Output:**

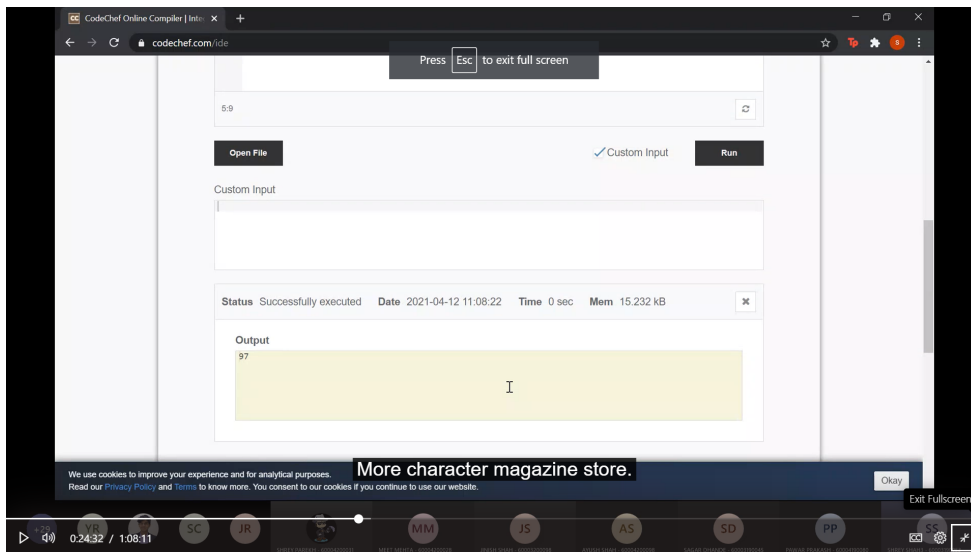


## Lecture 5: Break,Continue and Do-while

Date and Time: 12th April, 4:00 pm

In this lecture, the use of the important jump statements - break and continue is taught. A new type of loops - do while is introduced. Homework problems of the previous lectures were also discussed, This lecture was taken by Shrey and Rajvi, and around 35 students attended the lecture.

LINK: <https://web.microsoftstream.com/video/a4d5e136-5ec0-4b3b-b438-7da2314bcf67>

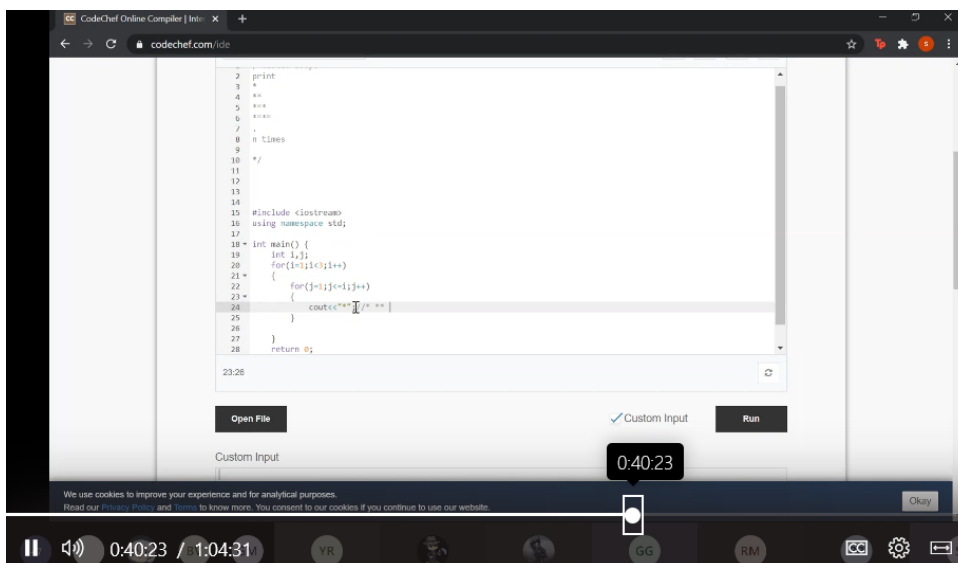


## Lecture 6: Nested loops

Date and Time: 19th April, 4:00 pm

In this lecture, how loops can be nested to write more complex code was taught. Easy pattern printing programs were taught. Homework from lecture 5 was also discussed which included trivial problems such as sum of digits, reversing a number, etc. This lecture was taken by Shrey and Parth.

LINK: <https://web.microsoftstream.com/video/9c98dfea-683e-467c-9deb-7a7746aedcd0>





## Contests

### 1. Contest #1

**Date and Time:** 11th October, 9pm.

**Top 3 Highest Scorers:**

1st: Ujjwal Prahladka

2nd: Sagar

3rd: Sakshi Uppoor

**Target Audience:** SEs and TEs/ SEs only

This contest covered important topics: Binary Search and Stacks. Competitors' ad-hoc, greedy, constructive and DSA skills were tested.

**Link:** <https://vjudge.net/contest/400424>

### 2. Contest #2

**Date and Time:** 1st November, 9pm.

**Top 3 Highest Scorers:**

1st: Ujjwal Prahladka

2nd: Aryan Parekh

3rd: Onkar

**Target Audience:** SEs and TEs/ SEs only

This contest covered problems on integer overflows, modular arithmetics, binary exponentiation, hashing and hashmaps. Competitors' ad-hoc, greedy, constructive and DSA skills were tested.

**Link:** <https://vjudge.net/contest/405113>

### 3. Code Uncode 4.0

Code Uncode is the annual flagship event of DJ Codestars, the official programming club of DJ Sanghvi College of Engineering. It is a 3 hour long intercollegiate coding contest with problems ranging in difficulty from easy to hard.

This year, due to the covid-19 pandemic, the event was conducted online on the platform codechef (<https://www.codechef.com/UNCO2021>). We had 7 problems, out of which only 2 people were able to solve the most difficult one. The contest was kept open to all and received 410 registrations.

Some popular codechef users like Taranpreet Singh ([https://www.codechef.com/users/taran\\_1407](https://www.codechef.com/users/taran_1407)) also participated. We also received participation from global users from countries like Azerbaijan, Ukraine and Canada. The top performers were from DA IICT, Gujarat and various IIT BHU.



The total prize pool this year is of 10,000 INR, which will be distributed to the top 3 Indian coders as follows:

1st Place: 5000 INR

2nd Place: 3000 INR

3rd Place: 2000 INR

We also have goodies from our sponsors that will be delivered to the top 3 Indian coders, along with the top 2 DJ Sanghvi Coders.

We are proud to announce the top 2 coders from DJ Sanghvi, Ujjwal from Third Year, Computer Engineering (Rank 19) and Parth Pawar from Second Year, Computer Engineering (Rank 28). Parth has also been a constant attendee of codestars lectures and is a part of the club giving FEs an introduction to programming.

The screenshot shows a mobile browser view of a CodeChef contest page. At the top, the URL is 'codechef.com/UNCO2021?it'. Below the header, there's a banner with a trophy and text: 'Try your hand at one of the competitions; solve a problem and write great code. Put yourself up for recognition and win great prizes. Contests begin on the first of every month.' Below this is a table with columns: Name, Code, Successful Submissions, and Accuracy. A message states 'Problems will be available in 18 hrs 7 mins 55 sec'. There are social media sharing options (Tweet, Like, Share) and an 'ANNOUNCEMENTS' section which is currently empty. The main content is titled 'CODE UNCODE 4.0 - DETAILS' and includes an 'About the Contest' section with bullet points: 'This is a coding contest based on algorithms, data structures and problem solving. If you are new here, check out our [getting started page](#).', 'Organiser: The contest is hosted by DJ Codestars, the official programming club of Dwarkadas J. Sanghvi College of Engineering.', 'Prizes: Total prizes worth INR 10K.', and 'Registrations for prizes: <https://www.meraevents.com/event/DJCodeUnicode>'. Below this is a 'Contest Details' section with: 'Duration: 3 hours', 'Start time: 11th March 2021, 15:00 hrs IST', and 'End time: 11th March 2021, 18:00 hrs IST. You may check your timezone [here](#).' The final section is 'Registration:', stating 'You need to have a CodeChef username to participate. If you do not have a CodeChef ID create one [here](#).'