

Pointing Subsequences whose sum is K



TUF

L7. All Kind of Patterns in Recursion | Print All | Print one | Count



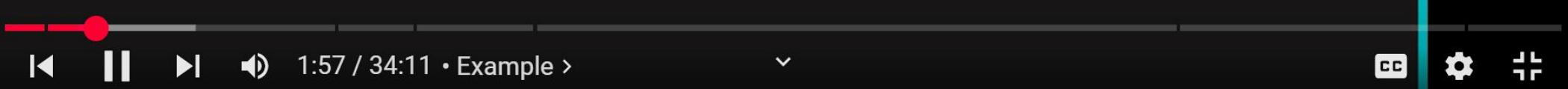
$arr \rightarrow \{1, 2, 1\}$

Sum = 2

$\{1, 1\}$
 $\{2\}$



TUF



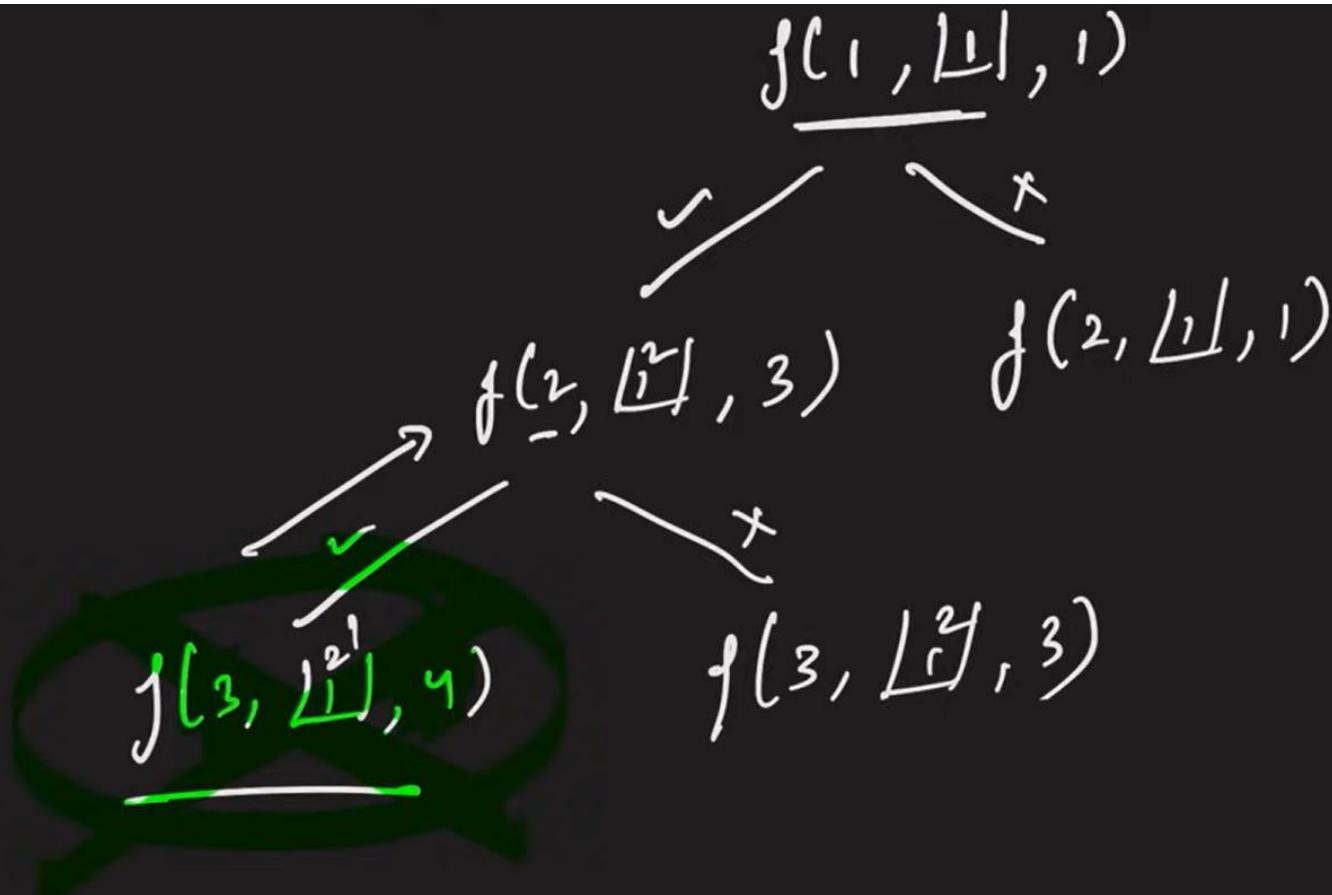
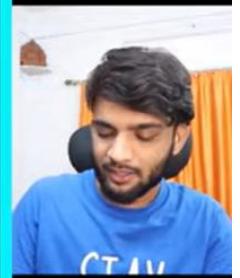
$$\begin{array}{ccc} f(0, \sqcup, 0) & & f(1, \sqcup, 0) \\ \swarrow \quad \searrow & & \rightarrow \\ f(1, \sqcup\sqcup, 1) & & \\ \swarrow \quad \searrow & & \\ f(2, \sqcup\sqcup, 3) & & f(2, \sqcup\sqcup, 1) \end{array}$$

l 2 s

0 1 2



TUF



$f()$

$f()$

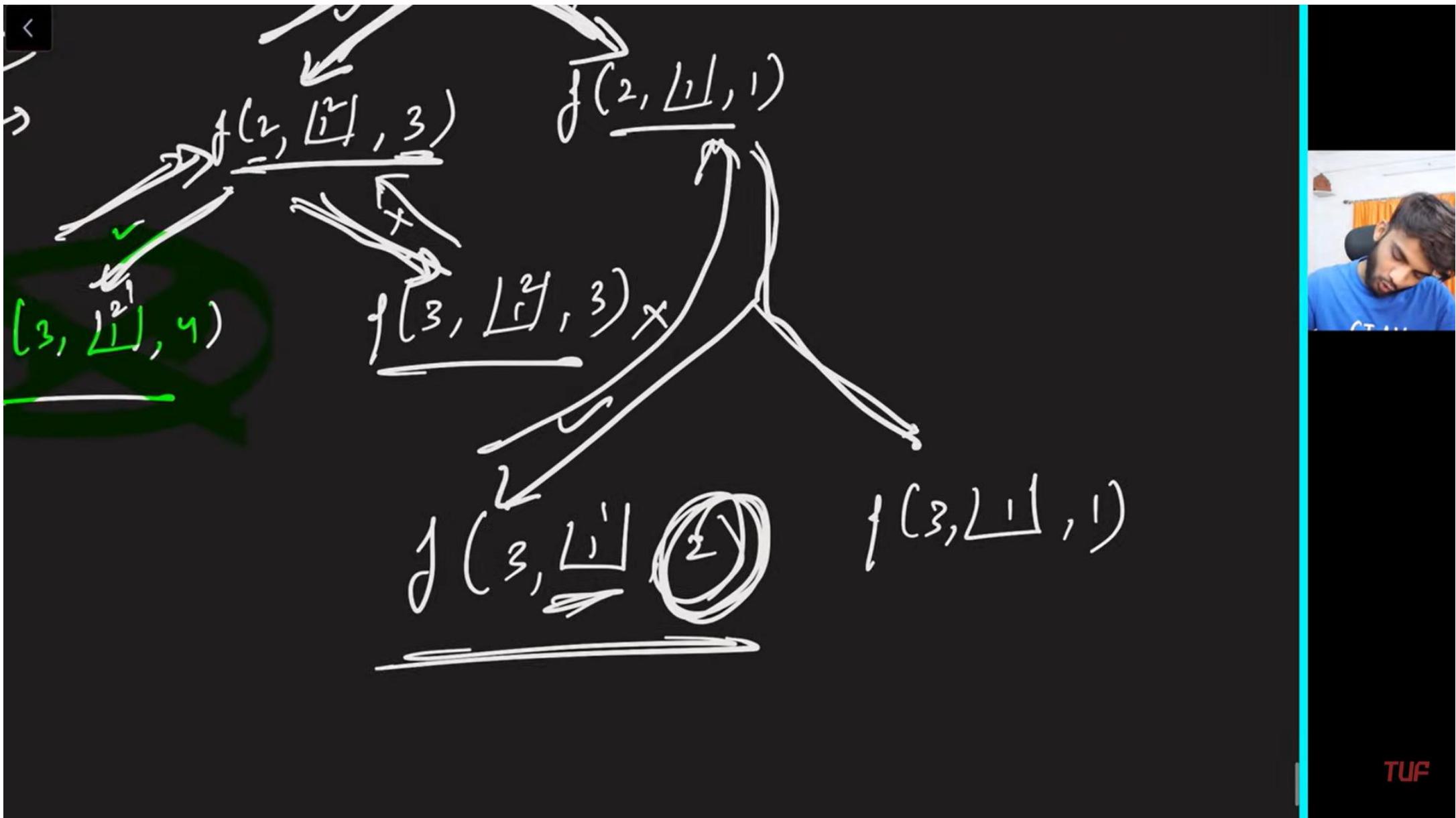
$f()$

$f(3, \underline{1}, 4)$

$f(1, \underline{1}, 1)$

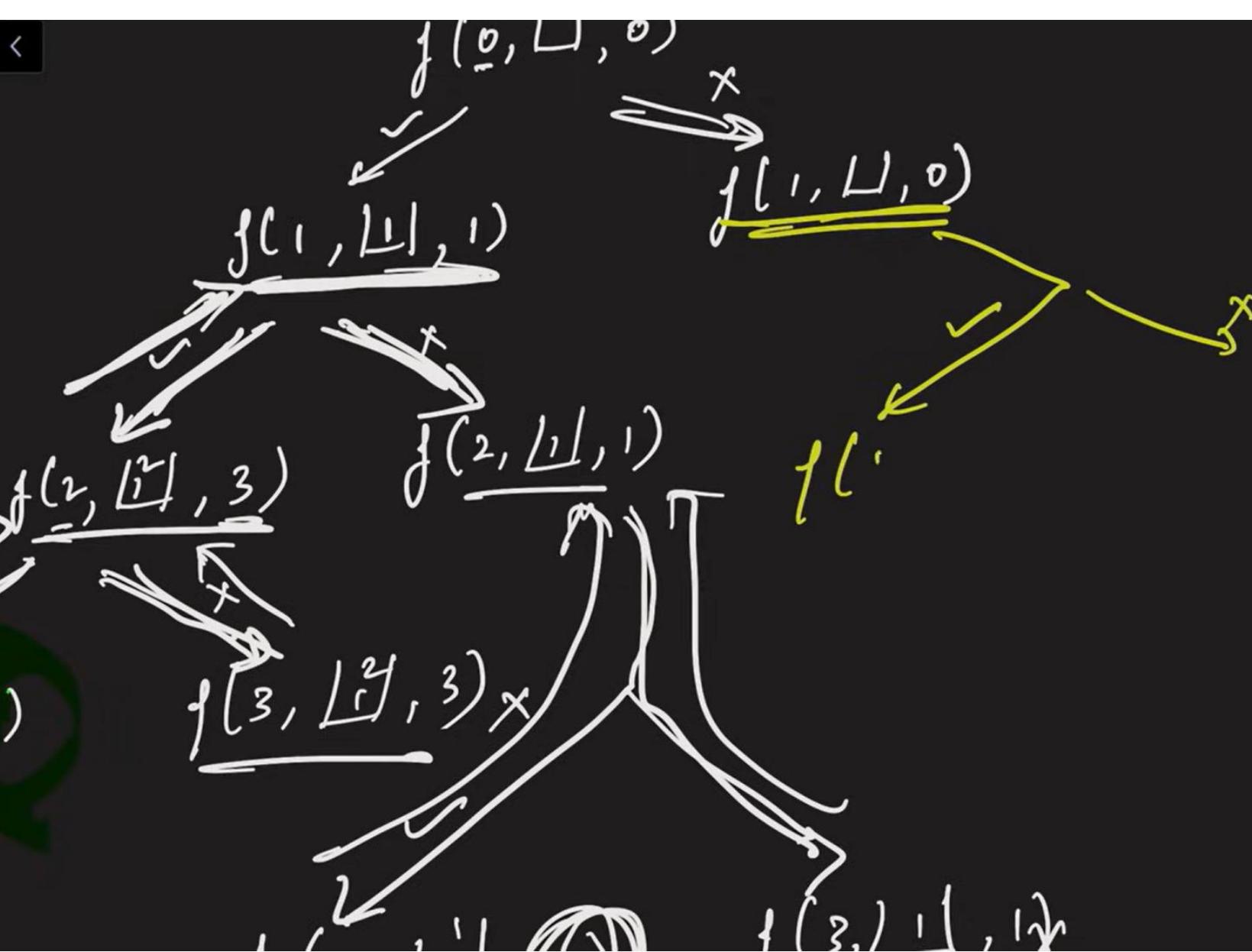
$f(2, \underline{1}, 1)$

$f(3, \underline{1}, 3) \times$

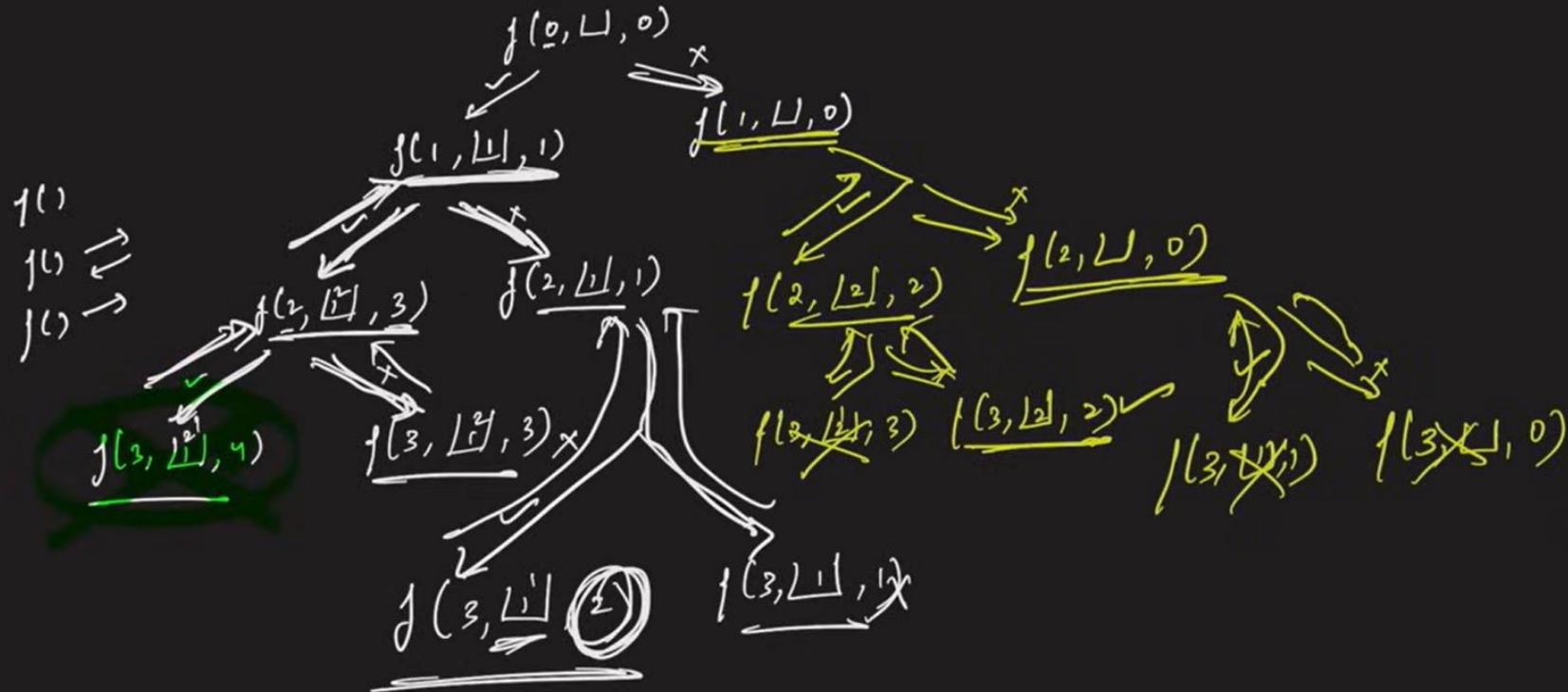




TUF



<

 $\begin{bmatrix} 1, 1 \\ 2 \end{bmatrix}$
 $\xrightarrow{\text{area}} \begin{bmatrix} 1, 2, 1 \\ 0, 1, 2 \end{bmatrix}$


$$\frac{1, 1}{2}$$

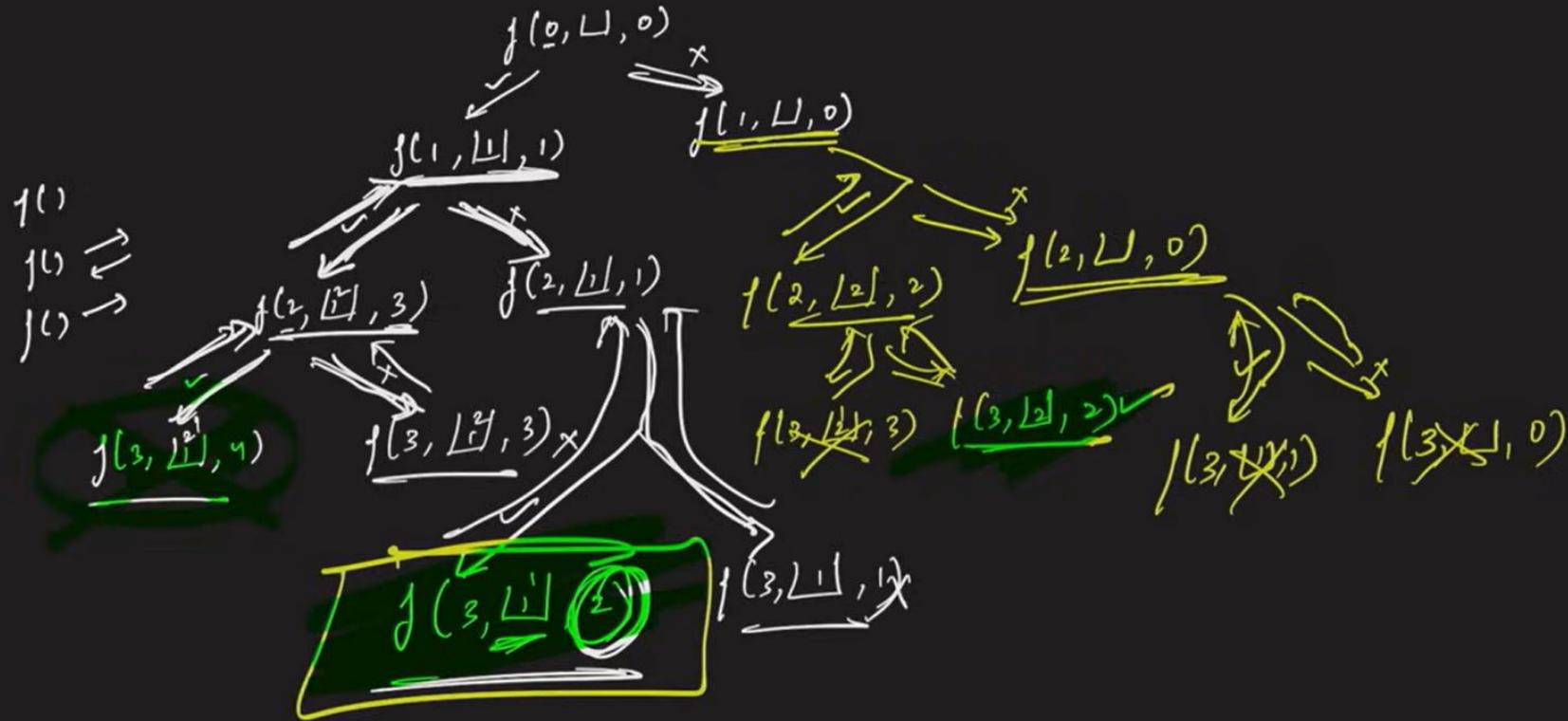


TUF

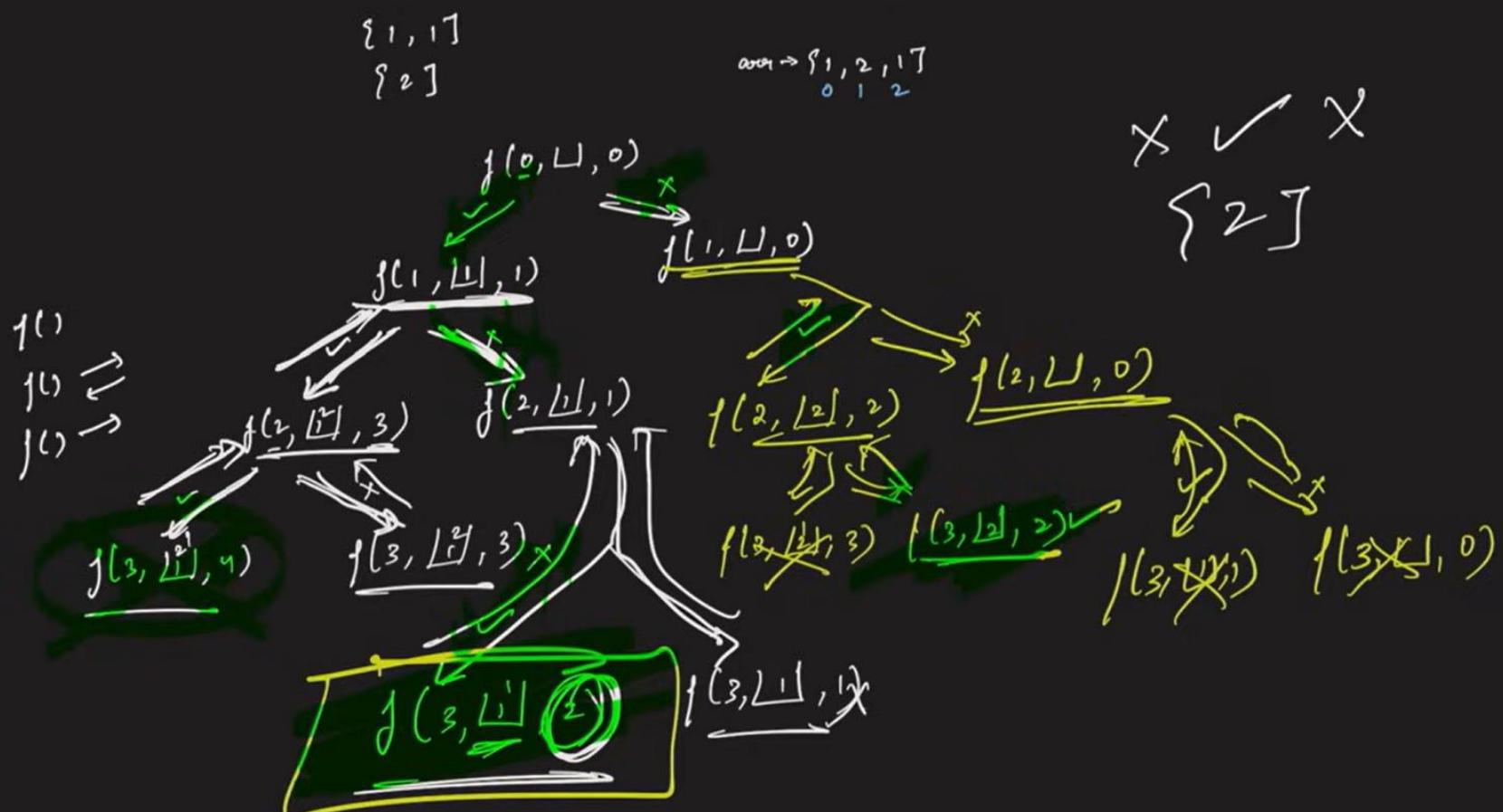
<

$$\begin{bmatrix} 1, 1 \\ 2 \end{bmatrix}$$

$$0 \rightarrow \begin{bmatrix} 1, 2, 1 \\ 0, 1, 2 \end{bmatrix}$$



$$\frac{1, 1}{2}$$



$\checkmark \times \checkmark$
 $\{1 \quad 1\}$

$$\frac{1, 1}{2}$$

```
<           f(i, L, s)
    {
        if (i == n)
            s = i (s == sum)
            point(ds)
            cout ;
        }
        ds.add (area [i]);
        S += area [i]
    }

    f (i + 1, "
```



`ds.add(cover[i]);`

$S+ = cover[i]$

$f(i+1, ds, S)$

`ds.remove(cover[i]);`

$S- = cover[i];$

ds



```
<           f(i, L, s)
{           y(i == n)
    s += y(s == sum)
    print(ds)
    auto;
}
ds.add(avr[s]);
s -= avr[s];
f(i+1, ds, s)
ds.remove(avr[s]);
s -= avr[s];
f(i+1, ds, s)
}
```



ds.add (cover S_i),

$S^+ = \text{cover } S_i]$

$f(i+1, ds, S)$

ds.remove (cover $S_i]$)

$S^- = \text{cover } S_i]$

$f(i, ds, S)$

TUF

dS . remove ($\text{arr } S[i]$)

$S \rightarrow \text{arr } S[i];$

$f(i+1, dS, S)$

}



```
code.cpp
```

```
1 #include<bits/stdc++.h>
2 using namespace std;
3 void printS(int ind, vector<int> &ds, int s, int sum, int arr[], int n) {
4     if(ind == n) {
5         if(s == sum) {
6             for(auto it : ds) cout << it << " ";
7             cout << endl;
8         }
9         return;
10    }
11
12    ds.push_back(arr)
13 }                                arr
14 int main() {                      auto
15 #ifndef ONLINE_JUDGE
16     freopen("input.txt", "r", stdin);
17     freopen("output.txt", "w", stdout);
18 #endif
19     int arr[] = {1, 2, 1};
20     int n = 3;
21     int sum = 2;
22     vector<int> ds;
23     printS(0, ds, 0, sum, arr, n);
24
25     return 0;
26 }
```

[Finished in 1.3s]

```
input.txt
```

```
1 5
2 1 2 3 4 5
```

```
output.txt
```

```
1 {}
2 2
3 1
4 1 2
5 3
6 3 2
7 3 1
8 3 1 2
9
```

TUF

```
code.cpp
```

```
4     if(ind == n) {
5         if(s == sum) {
6             for(auto it : ds) cout << it << " ";
7             cout << endl;
8         }
9         return;
10    }
11
12    ds.push_back(arr[ind]);
13    s += arr[ind];
14
15    printS(ind+1, ds, s, sum, arr, n);
16
17    s-= arr[ind];
18    ds.pop_back();
19
20    // not pick
21    printS(ind+1,ds, s, sum, arr, n);
22}
23 int main() {
24 #ifndef ONLINE_JUDGE
25     freopen("input.txt", "r", stdin);
26     freopen("output.txt", "w", stdout);
27 #endif
28     int arr[] = {1, 2, 1};
29     int n = 3;
30     int sum = 2;
31     vector<int> ds;
32     printS(0, ds, 0, sum, arr, n);
33
34     return 0;
35 }
```

[Finished in 1.3s]

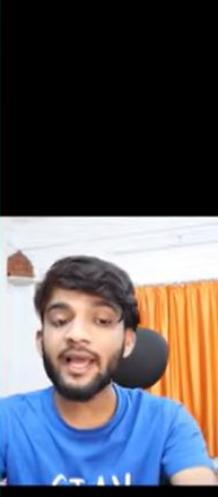
```
input.txt
```

```
1 5
2 1 2 3 4 5
```

```
output.txt
```

```
1 {}
2 2
3 1
4 1 2
5 3
6 3 2
7 3 1
8 3 1 2
9
```





```
code.cpp      stdc++.h
1 #include<bits/stdc++.h>
2 using namespace std;
3 void printS(int ind, vector<int> &ds, int s, int sum, int arr[], int n) {
4     if(ind == n) {
5         if(s == sum) {
6             for(auto it : ds) cout << it << " ";
7             cout << endl;
8         }
9         return;
10    }
11
12    ds.push_back(arr[ind]);
13    s += arr[ind];
14
15    printS(ind+1, ds, s, sum, arr, n);
16
17    s -= arr[ind];
18    ds.pop_back();
19
20    // not pick
21    printS(ind+1,ds, s, sum, arr, n);
22 }
23 int main() {
24     #ifndef ONLINE_JUDGE
25     freopen("input.txt", "r", stdin);
26     freopen("output.txt", "w", stdout);
27     #endif
28     int arr[] = {1, 2, 1};
29     int n = 3;
30     int sum = 2;
31     vector<int> ds;
32     printS(0, ds, 0, sum, arr, n);
33
34     return 0;
35 }
```

[Finished in 1.9s]

input.txt

```
1 5
2 1 2 3 4 5
```

output.txt

```
1 1 1
2 2
3
```

TUF

```
code.cpp      stdc++.h
1 #include<bits/stdc++.h>
2 using namespace std;
3 void printS(int ind, vector<int> &ds, int s, int sum, int arr[], int n) {
4     if(ind == n) {
5         if(s == sum) {
6             for(auto it : ds) cout << it << " ";
7             cout << endl;
8         }
9         return;
10    }
11
12    ds.push_back(arr[ind]);
13    s += arr[ind];
14
15    printS(ind+1, ds, s, sum, arr, n);
16
17    s -= arr[ind];
18    ds.pop_back();
19
20    // not pick
21    printS(ind+1, ds, s, sum, arr, n);
22}
23 int main() {
24     #ifndef ONLINE_JUDGE
25     freopen("input.txt", "r", stdin);
26     freopen("output.txt", "w", stdout);
27     #endif
28     int arr[] = {1, 2, 1};
29     int n = 3;
30     int sum = 2;
31     vector<int> ds;
32     printS(0, ds, 0, sum, arr, n);
33
34     return 0;
35 }
```

[Finished in 1.9s]

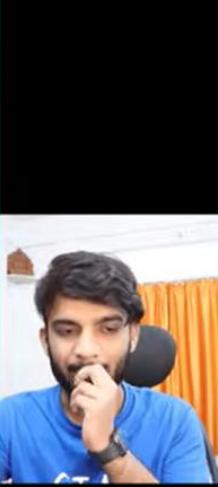
input.txt

```
1 5
2 1 2 3 4 5
```

output.txt

```
1 1 1
2 2
3
```





A screenshot of a code editor and terminal window. The code editor shows a C++ file named 'code.cpp' with the following content:

```
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 bool flag = false;
5 void printS(int ind, vector<int> &ds, int s, int sum, int arr[], int n) {
6     if(ind == n) {
7         if(s == sum && flag == false) {
8             flag = true;
9             for(auto it : ds) cout << it << " ";
10            cout << endl;
11        }
12        return;
13    }
14
15    ds.push_back(arr[ind]);
16    s += arr[ind];
17
18    printS(ind+1, ds, s, sum, arr, n);
19
20    s -= arr[ind];
21    ds.pop_back();
22
23    // not pick
24    printS(ind+1, ds, s, sum, arr, n);
25}
26 int main() {
27     #ifndef ONLINE_JUDGE
28         freopen("input.txt", "r", stdin);
29         freopen("output.txt", "w", stdout);
30     #endif
31     int arr[] = {1, 2, 1};
32     int n = 3;
33     int sum = 2;
34     vector<int> ds;
35     printS(0, ds, 0, sum, arr, n);
36 }
```

The terminal window shows the input file 'input.txt' containing:

```
1 5
2 1 2 3 4 5
```

The terminal window shows the output file 'output.txt' containing:

```
1 1 1
2
```

[Finished in 1.9s]

Print any 1 subsequence where S_m is Sum.

$f()$

{

$Tf()$

True

X $f()$

?

TUF

Subsequence where $\underline{S_m}$ is Sum.

(Technique to print
One answer)

$f()$

{

$f()$ → True

X $f()$ →

?

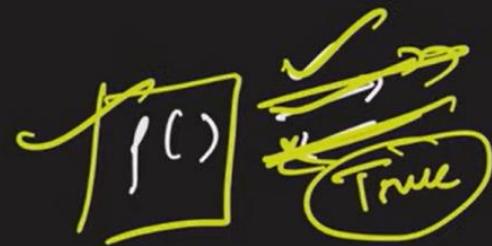


TUF

<

0

{



3

f()

{

base casecond \rightarrow satisfied

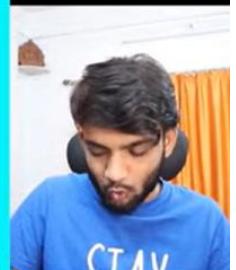
out (true)

 $\rightarrow x$ sat

return false

$$\text{if } (f() == \text{true})$$

true



TUF

< any 1 subsequence whose sum is sum:

(Technique to print
One answer)

$f()$

{

~~$f()$~~ ✓
true

X ~~$f()$~~ ↗
?

$f()$

{

base case

cond → satisfied
out (true)
→ X sat
sub false

if ($f()$ == true)
return;

$f()$

TUF

```
code.cpp • stdc++.h
1 #include<bits/stdc++.h>
2 using namespace std;
3
4
5 bool printS(int ind, vector<int> &ds, int s, int sum, int arr[], int n) {
6     if(ind == n) {
7         // condition satisfied
8         if(s == sum) {
9             for(auto it : ds) cout << it << " ";
10            cout << endl;
11            return true;
12        }
13        // condition not satisfied
14        else return false;
15    }
16
17    ds.push_back(arr[ind]);
18    s += arr[ind];
19
20    if(printS(ind+1, ds, s, sum, arr, n) == true) {
21        return true;
22    }
23
24    s -= arr[ind];
25    ds.pop_back();
26
27    // not pick
28    if(printS(ind+1, ds, s, sum, arr, n) == true) return true;
29
30    return false;
31 }
32 int main() {
33     #ifndef ONLINE_JUDGE
34     freopen("input.txt", "r", stdin);
35     freopen("output.txt", "w", stdout);
36     #endif
}
[Finished in 1.9s]
```

```
input.txt
1 5
2 1 2 3 4 5
output.txt
1 1 1
2
```



TUF

```
code.cpp  stdc++.h
9      for (auto it : ds) cout << it << " ";
10     cout << endl;
11     return true;
12 }
13 // condition not satisfied
14 else return false;
15 }
16
17 ds.push_back(arr[ind]);
18 s += arr[ind];
19
20 if (printS(ind+1, ds, s, sum, arr, n) == true) {
21     return true;
22 }
23
24 s -= arr[ind];
25 ds.pop_back();
26
27 // not pick
28 if (printS(ind+1, ds, s, sum, arr, n) == true) return true;
29
30 return false;
31 }
32 int main() {
33 #ifndef ONLINE_JUDGE
34     freopen("input.txt", "r", stdin);
35     freopen("output.txt", "w", stdout);
36 #endif
37     int arr[] = {1, 2, 1};
38     int n = 3;
39     int sum = 2;
40     vector<int> ds;
41     printS(0, ds, 0, sum, arr, n);
42
43     return 0;
44 }
```

[Finished in 2.2s]

```
input.txt
1 5
2 1 2 3 4 5
```

```
output.txt
1 1
2
```



TUF

<

$$S_m = 2 \quad [1, 2, 1]$$

$f(0, \sqcup, 0)$
{
 x

de.add()
 $y(f(1, \sqcup, 1))$

ds.remove

S -

$f(1, \sqcup, 0))$

TUF

<

$$\delta_m = 2$$

$$\{1, 2, 1\}$$

$$f(1, \boxed{1}, 1)$$

$$iy(\cdot)x$$

ds.add

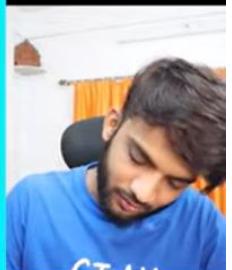
$$iy(f[\boxed{2}, \boxed{1}], 3)$$

$$f(2, \boxed{1}, 3)$$

$$iy(\)x$$

ds.add

$$iy(f[$$



$\{1, 2, 1\}$

1)

x

y, 3)

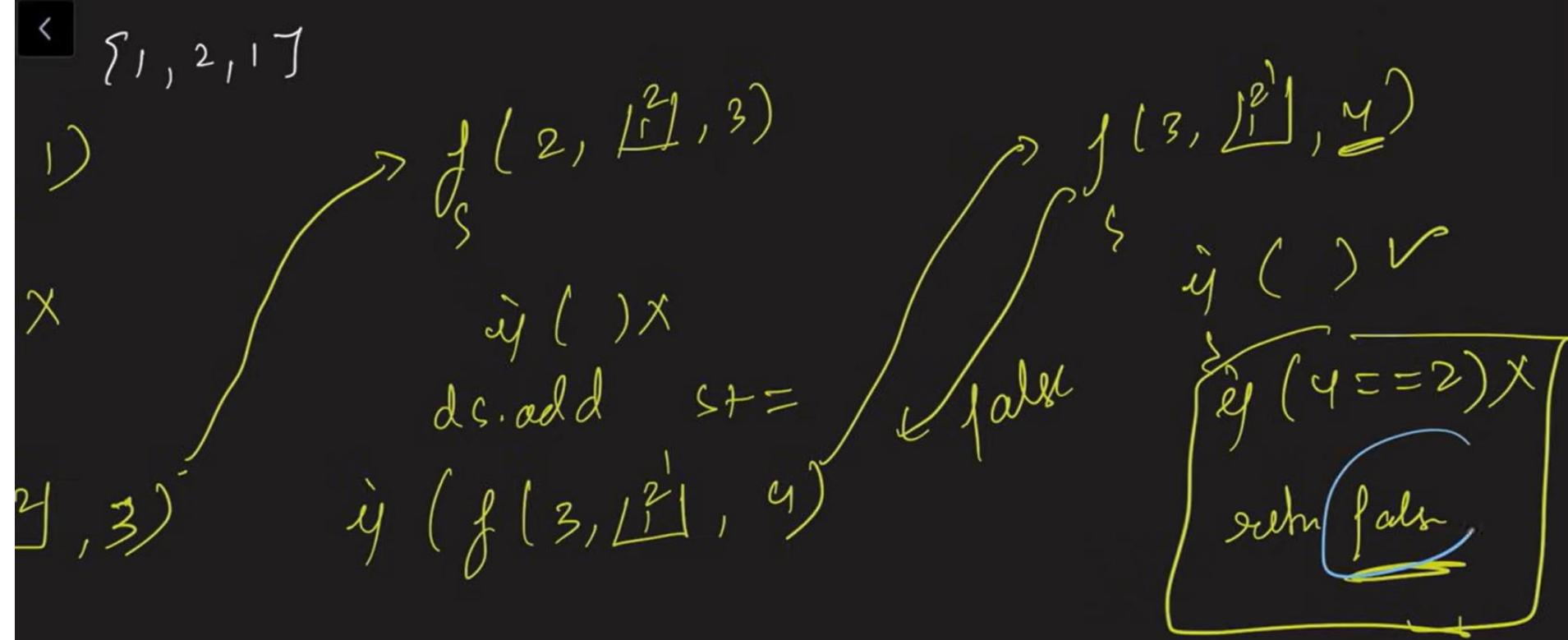
$f(2, \boxed{1}, 3)$

$\bar{y}(\)x$

ds.add s+=

$\bar{y}(f(3, \boxed{1}, 4))$

TUF



TUF

L7. All Kind of Patterns in Recursion | Print All | Print one | Count

→ Count the Subsequences with sum = 12

{1, 2, 1}

k=2

(2 subsequences)



23:57 / 34:11 • Modify >



TUF

L7. All Kind of Patterns in Recursion | Print All | Print one | Count

→ Count the Subsequences with sum = 12

[1, 2, 1]

k=2

(2 subsequences)



TUF



23:52 / 34:11 • Modify >



```
int f()
```

```
{
```

base case

subn 1 → condition satisfies

subn 0 → condition not satisfied

```
l = f()
```

```
r = f()
```

subn l+r;

```
}
```



```
int f()
```

```
{
```

base case

return 1 → condition satisfies

return 0 → condition not satisfied

```
l = f()
```

```
r = f()
```

return l+r;

```
}
```



TUF

{

base case

subn 1 → condition satisfies

subn 0 → condition not satisfied

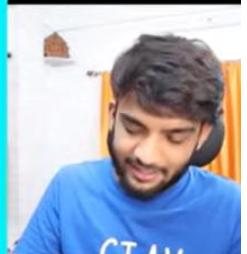
$$\begin{cases} l = f() \\ r_1 = f() \end{cases}$$

subn $l+r_1$;

}

$$\begin{aligned} s &= 0 \\ fn(i &\xrightarrow{i \rightarrow n}) \\ s+ &= f(); \end{aligned}$$

subn s



{

base case

subn 1 → condition satisfies

subn 0 → condit n catrs fine

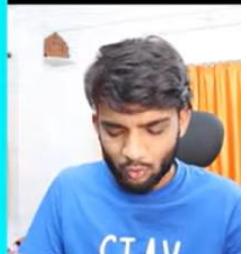
$$\left\{ \begin{array}{l} l = f() \\ r_1 = j() \end{array} \right. ^2$$

subn $l+r_1;$

}

$$\begin{aligned} s &= 0 \\ f_n(i) &\quad i \mapsto n \\ s+ &= f() \end{aligned}$$

subn s



{

base case

subn 1 → condition satisfies

subn 0 → condit n catrs fine

$$\left\{ \begin{array}{l} l = f() \\ r_1 = j() \end{array} \right.$$

2

}

$s = 0$

$f_n(i \in 1 \rightarrow n)$

$s + = f()$

$\} N \text{ Outer}$

subn s

```
code.cpp stdc++.h
5 int printS(int ind, vector<int> &ds, int s, int sum, int arr[], int n) {
6     if(ind == n) {
7         // condition satisfied
8         if(s == sum) {} else return 0;
9         return 1;
10    }
11    // condition not satisfied
12    else return 0;
13 }
14
15 ds.push_back(arr[ind]);
16 s += arr[ind];
17
18 int l = printS(ind+1, ds, s, sum, arr, n) ;
19
20 s-= arr[ind];
21 ds.pop_back();
22
23 // not pick
24 int r = printS(ind+1,ds, s, sum, arr, n);
25
26 return l + r;
27 }
28 int main() {
29 #ifndef ONLINE_JUDGE
30     freopen("input.txt", "r", stdin);
31     freopen("output.txt", "w", stdout);
32 #endif
33     int arr[] = {1, 2, 1};
34     int n = 3;
35     int sum = 2;
36     vector<int> ds;
37     printS(0, ds, 0, sum, arr, n);
38
39     return 0;
}
[Finished in 2.2s]
```

input.txt

```
1 5
2 1 2 3 4 5
```

output.txt

```
1 1 1
2
```

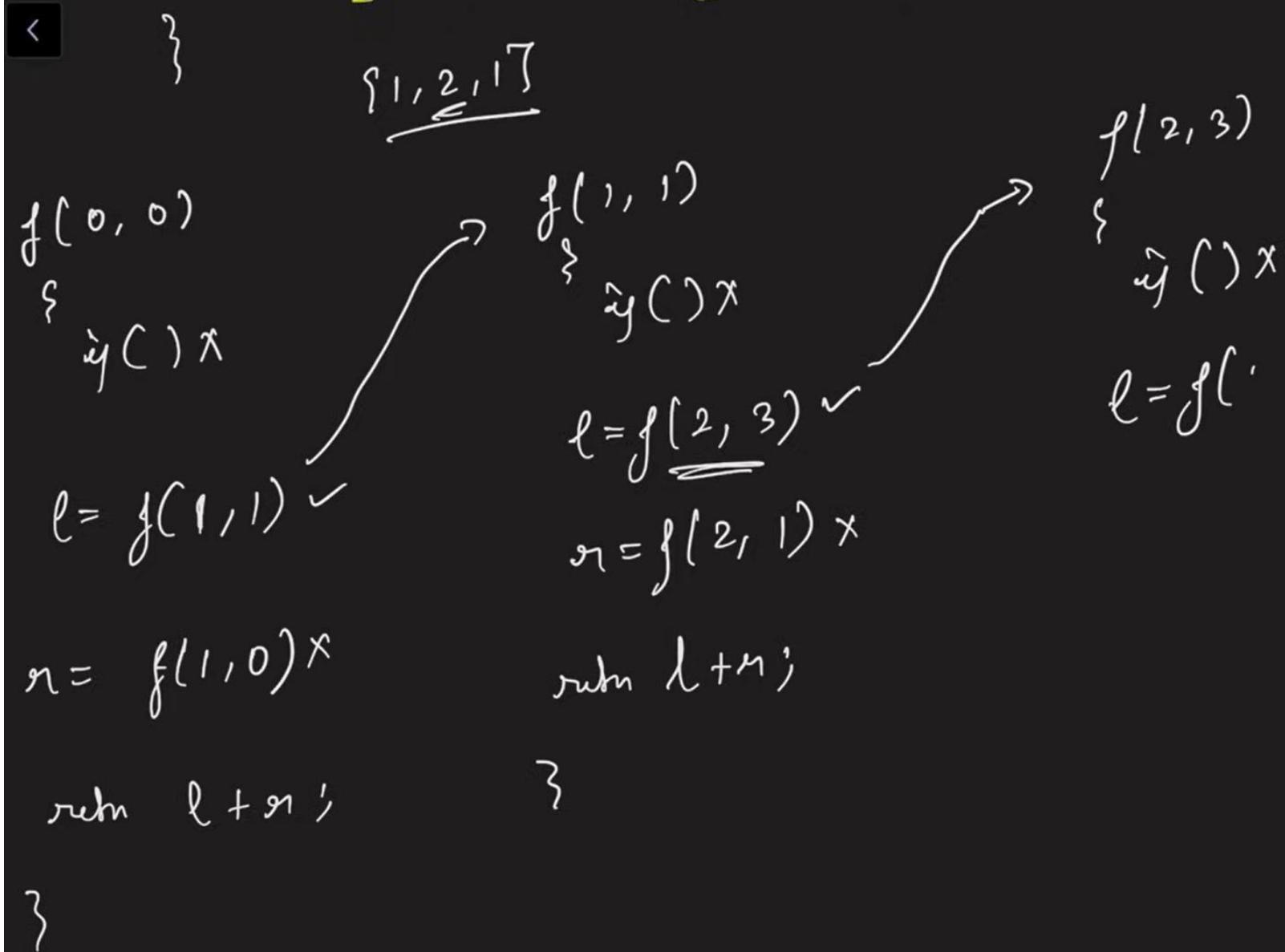


TUF

```
code.cpp x stdc++.h
1 #include <iostream>
2 using namespace std;
3
4 int printS(int ind, int s, int sum, int arr[], int n) {
5     if(ind == n) {
6         // condition satisfied
7         if(s == sum) return 1;
8         // condition not satisfied
9         else return 0;
10    }
11
12    s += arr[ind];
13
14    int l = printS(ind+1, s, sum, arr, n) ;
15
16    s-= arr[ind];
17
18    // not pick
19    int r = printS(ind+1, s, sum, arr, n);
20
21    return l + r;
22}
23
24 int main() {
25     #ifndef ONLINE_JUDGE
26         freopen("input.txt", "r", stdin);
27         freopen("output.txt", "w", stdout);
28     #endif
29     int arr[] = {1, 2, 1};
30     int n = 3;
31     int sum = 2;
32     cout << printS(0, 0, sum, arr, n);
33
34     return 0;
35 }
36
37 }
```

[Finished in 1.5s]

Line 19, Column 5; Reloading ~/Desktop/SublimeFolder/output.txt Tab Size: 4 C++



$f(1, 1) \checkmark$

$f(1, 0) \times$

$\ell + m;$

$$\ell = f(\underline{2}, \underline{3}) \checkmark$$

$$m = f(\underline{2}, 1) \times$$

run $\ell + m;$

}

$$\ell = \cancel{f(\underline{2}, \underline{3})}$$

$$m = f(\underline{2}, \cancel{3}) \times$$

run $\ell + m;$

3

$$f(2, 1)$$

\hat{y}^1



```
int f()
```

base case

subn 1 → condition satisfies

subn 0 → condition not satisfied

$$\left\{ \begin{array}{l} l = f() \\ r_1 = j() \end{array} \right. ^2$$

subn $l+r_1$

$$\left. \begin{array}{l} s = 0 \\ f_n(i) \rightarrow n \\ l+ = f() \end{array} \right\} N \text{ Power}$$

subn s



f(2, 3)

TUF



```
int f()
```

base case

sub 1 → condition satisfies

sub 0 → condition not satisfied

{
 l = f()
 r = j()
}

sub l+r;

$s = 0$
 $f_n(i \rightarrow n)$
 $s+ = f()$

N times

sub s

✓ { 1 } x { 1 } { 1, 2, 1 }

f(2, 3)

TUF

```
int f()
```

base case

subn 1 → condition satisfies

subn 0 → condition not satisfied

{
 l = f()
 r = j()
}

subn l+r

s = 0

f(n)

i = 1 → n

s +

= f()

i

N times

subn s

✓ { 1 } × { 1 } { 1, 2, 1 }

f(2, 3)

TUF

$$2^{\frac{2}{1}} = 2^2 \leq \approx 2^n$$



TUF

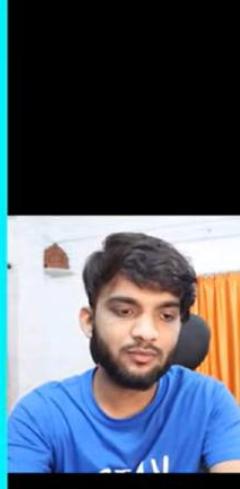
```
code.cpp      stdc++.h
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  int printS(int ind, int s, int sum, int arr[], int n) {
7
8      // condition not satisfied
9      if(s > sum) return 0;
10     if(ind == n) {
11         // condition satisfied
12         if(s == sum) return 1;
13         // condition not satisfied
14         else return 0;
15     }
16
17     s += arr[ind];
18
19     int l = printS(ind+1, s, sum, arr, n) ;
20
21     s-= arr[ind];
22
23
24     // not pick
25     int r = printS(ind+1, s, sum, arr, n);
26
27     return l + r;
28 }
29 int main() {
30     #ifndef ONLINE_JUDGE
31     freopen("input.txt", "r", stdin);
32     freopen("output.txt", "w", stdout);
33     #endif
34     int arr[] = {1, 2, 1};
35     int n = 3;
36     int sum = 2;
37 }
[Finished in 0.8s]
```

input.txt

1

output.txt

1 2



TUF

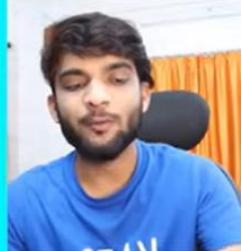
```
code.cpp stdc++.h
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4
5 using namespace std;
6
7 int printS(int ind, int s, int sum, int arr[], int n) {
8
9     // condition not satisfied
10    // strictly done if array contains positives only
11    if(s > sum) return 0;
12
13    if(ind == n) {
14        // condition satisfied
15        if(s == sum) return 1;
16        // condition not satisfied
17        else return 0;
18    }
19
20    s += arr[ind];
21
22    int l = printS(ind+1, s, sum, arr, n);
23
24    s -= arr[ind];
25
26    // not pick
27    int r = printS(ind+1, s, sum, arr, n);
28
29    return l + r;
30}
31 int main() {
32     #ifndef ONLINE_JUDGE
33     freopen("input.txt", "r", stdin);
34     freopen("output.txt", "w", stdout);
35     #endif
36     int arr[] = {1, 2, 1};
37     int n = 3;
38     int sum = 2;
39
40     cout << printS(0, 0, sum, arr, n) << endl;
41
42     return 0;
43 }
```

[Finished in 0.8s]

Line 8, Column 55

Tab Size: 4 C++

TUF



```
code.cpp stdc++.h
1 using namespace std;
2
3
4
5 int printS(int ind, int s, int sum, int arr[], int n) {
6
7     // condition not satisfied
8     // strictly done if array contains positives only
9     if(s > sum) return 0;
10
11    if(ind == n) {
12        // condition satisfied
13        if(s == sum) return 1;
14        // condition not satisfied
15        else return 0;
16    }
17
18    s += arr[ind];
19
20    int l = printS(ind+1, s, sum, arr, n);
21
22    s -= arr[ind];
23
24
25    // not pick
26    int r = printS(ind+1, s, sum, arr, n);
27
28    return l + r;
29}
30
31 int main() {
32     #ifndef ONLINE_JUDGE
33     freopen("input.txt", "r", stdin);
34     freopen("output.txt", "w", stdout);
35     #endif
36     int arr[] = {1, 2, 1};
37     int n = 3;
}
[Finished in 0.8s]
```

input.txt

1

output.txt

1 2



TUF

21 characters selected

Tab Size: 4

C++

$$\sum_{i=1}^n \frac{1}{i^2} = \frac{\pi^2}{6}$$

point → theorem

point 1 → cut T/F &
avoid fuction
Recursion calls
if you get true

P

2 2 2 2 2 $\approx (2)$

point \rightarrow theorem

point 1

\rightarrow not T/F &
avoid function

Recursion calls
if you get true

Cont

\rightarrow

not 1
not 0

TUF

point → screen

point |

→ not T/F &
avoid fractn

Recursion calls

if you get true

Cont

→

not
not 0

add all f()
{ endf }