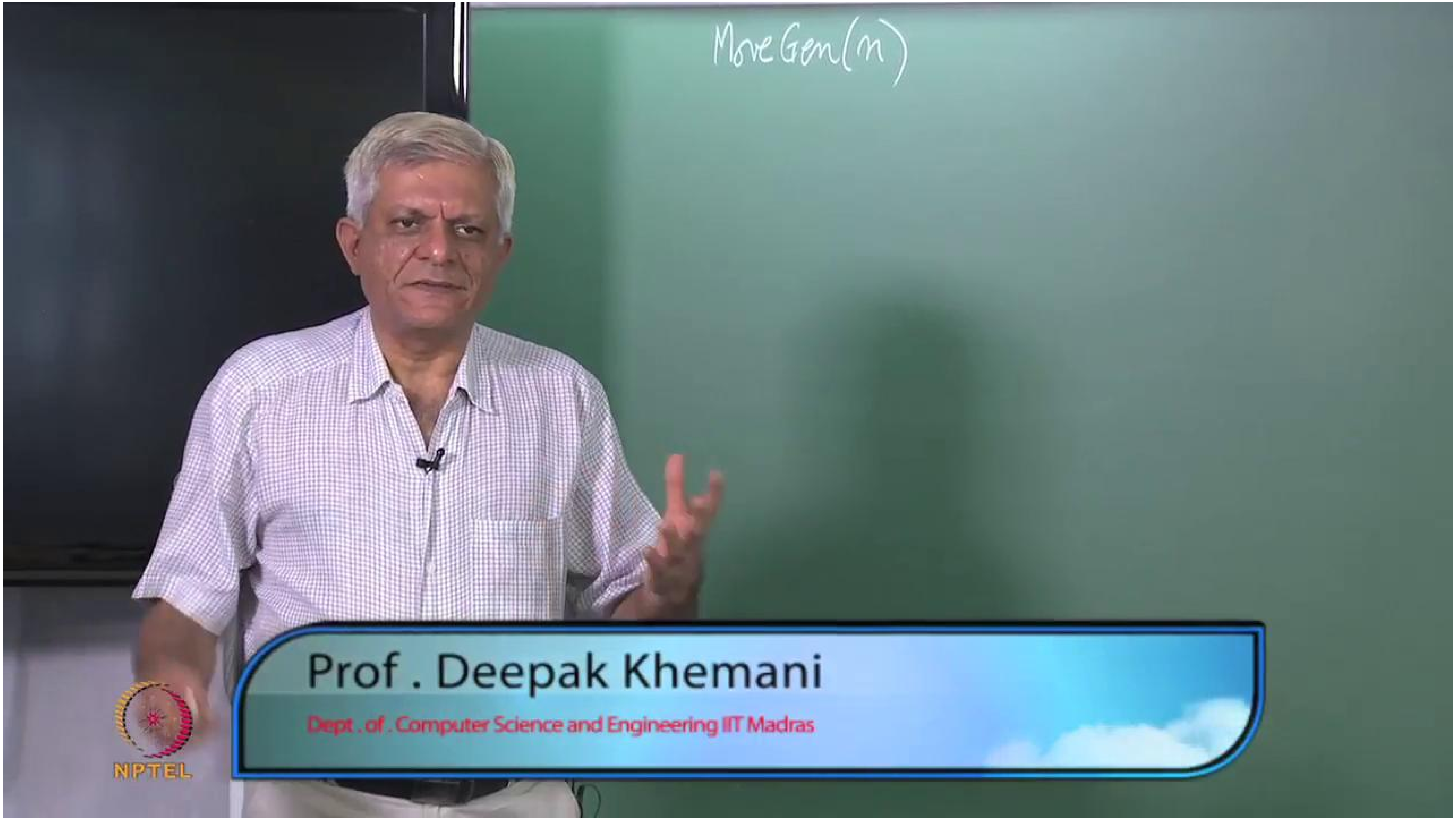
Video Lectures On

Artificial Intelligence

Lecture 08
Search DFID

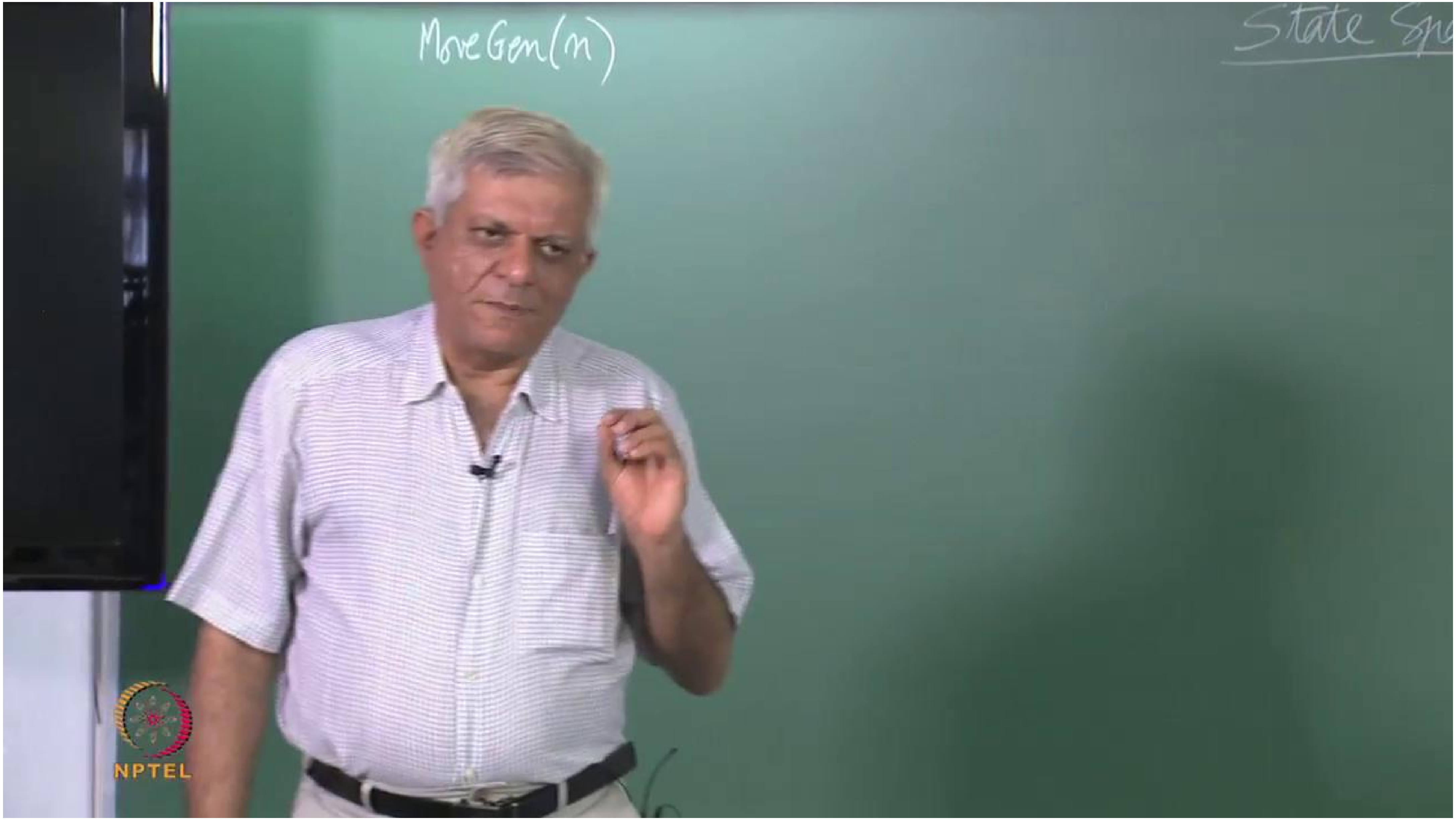
Prof. Deepak Khemani
Department of Computer Science and Engineering
IIT Madras

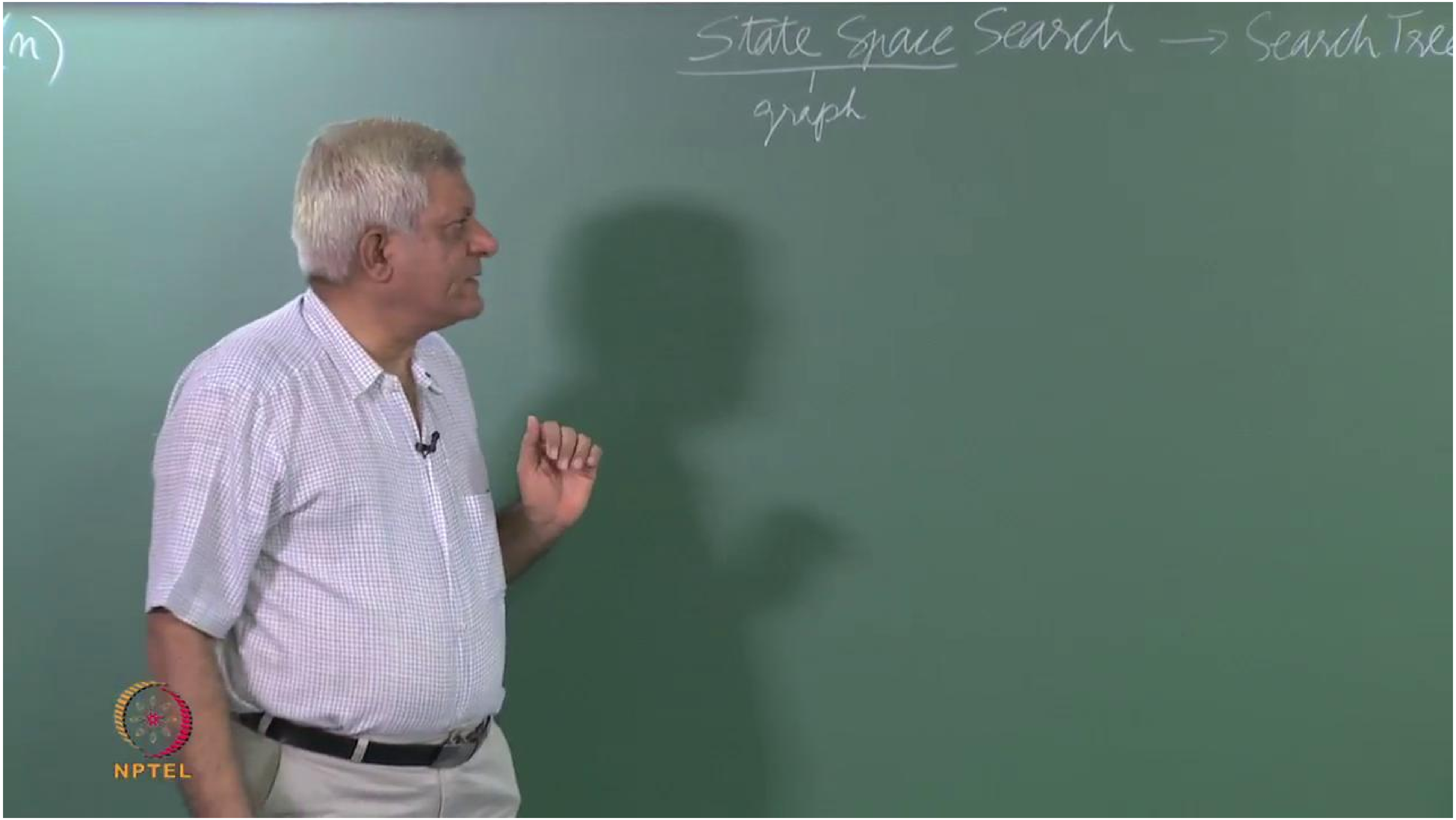
NPTEL



Prof . Deepak Khemani

Dept . of . Computer Science and Engineering IIT Madras

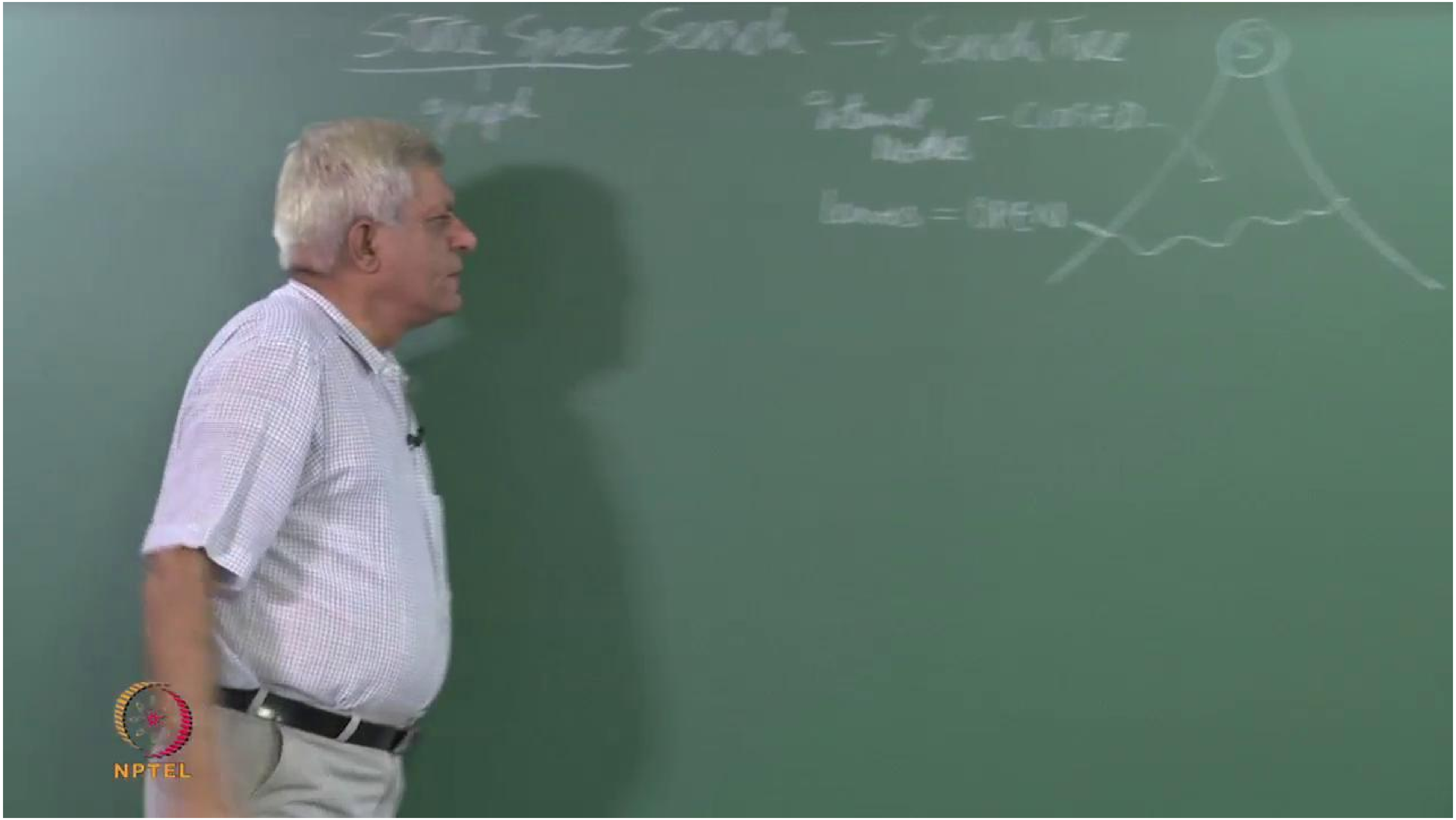

MoveGen(n)
State Spa



(n)

State Space Search → Search Tree

graph

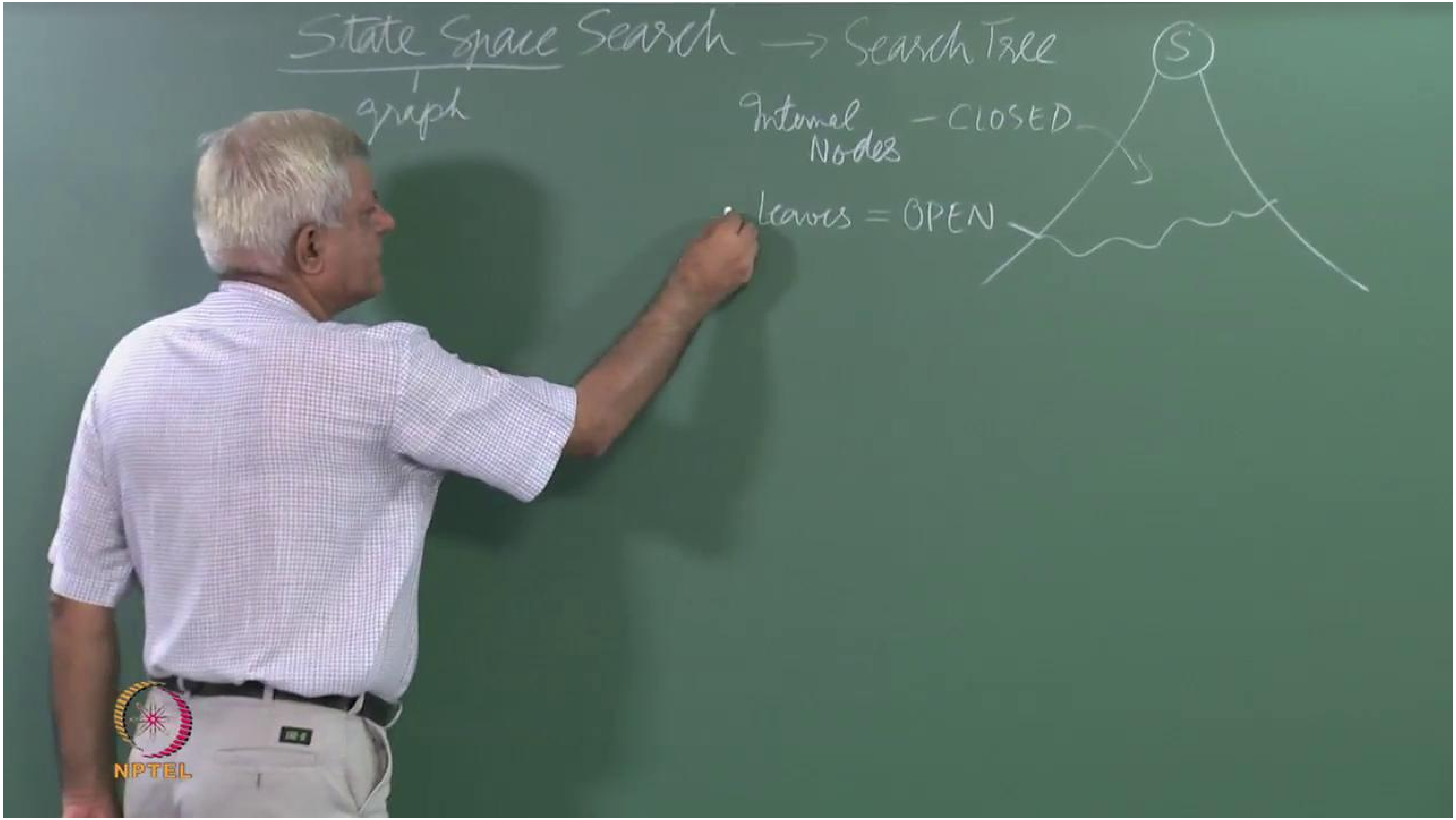


State Space Search → Search Tree

graph

Visited Nodes – CLOSED

Leaves = OPEN

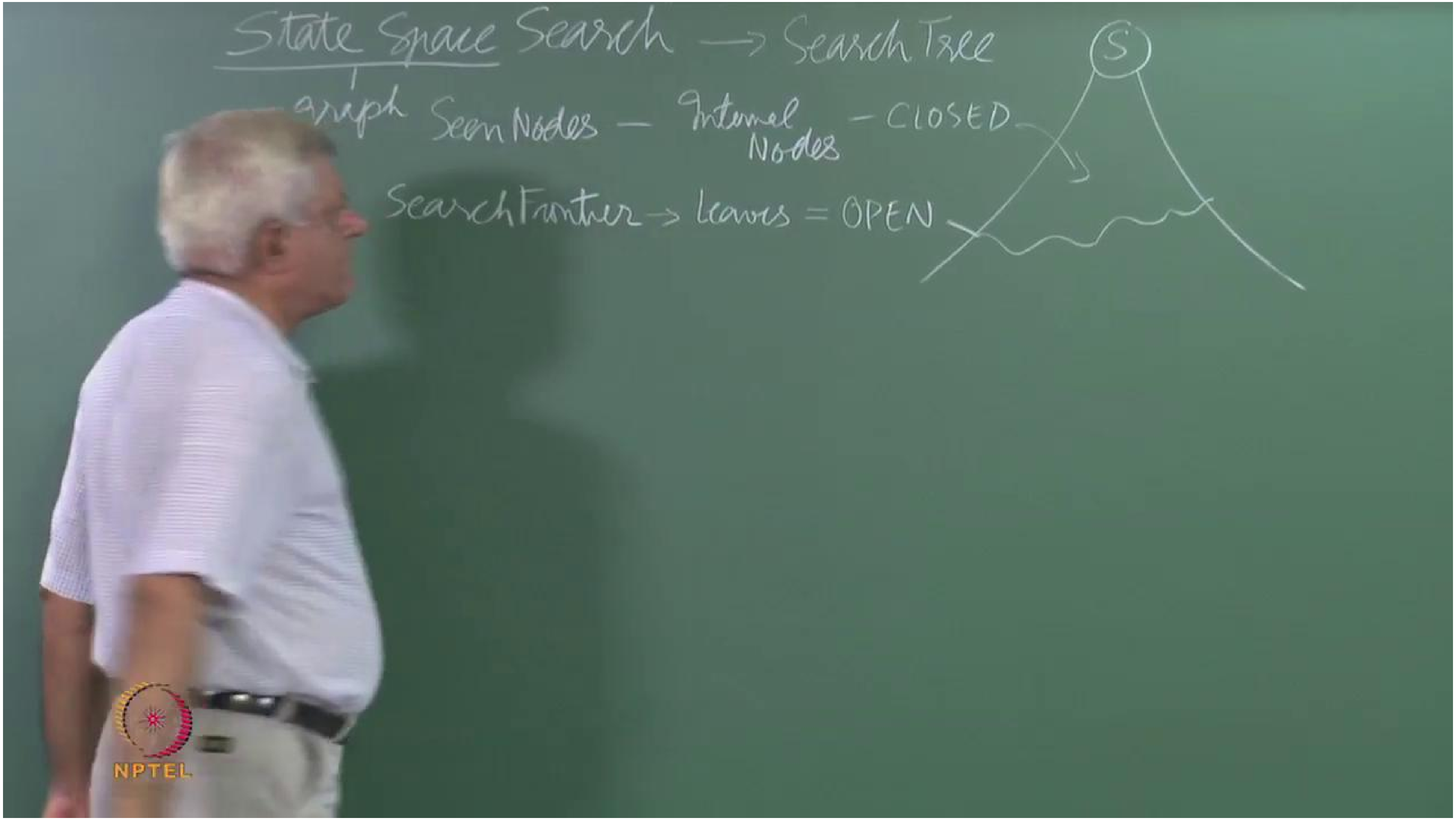


State Space Search → Search Tree

graph

Internal Nodes – CLOSED

Leaves = OPEN

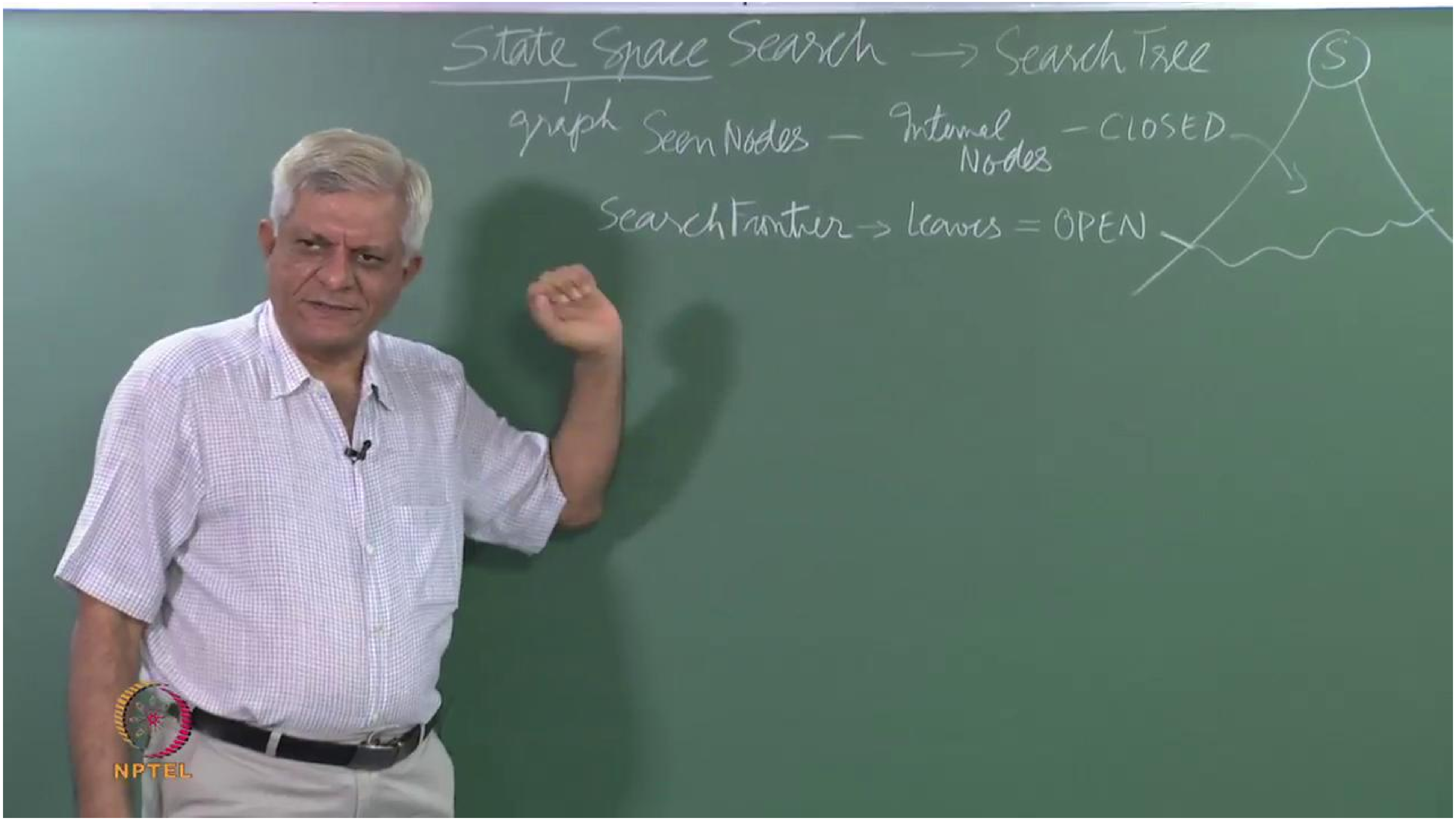


State Space Search → Search Tree

graph

Seen Nodes — Internal Nodes — CLOSED

Search Further → Leaves = OPEN

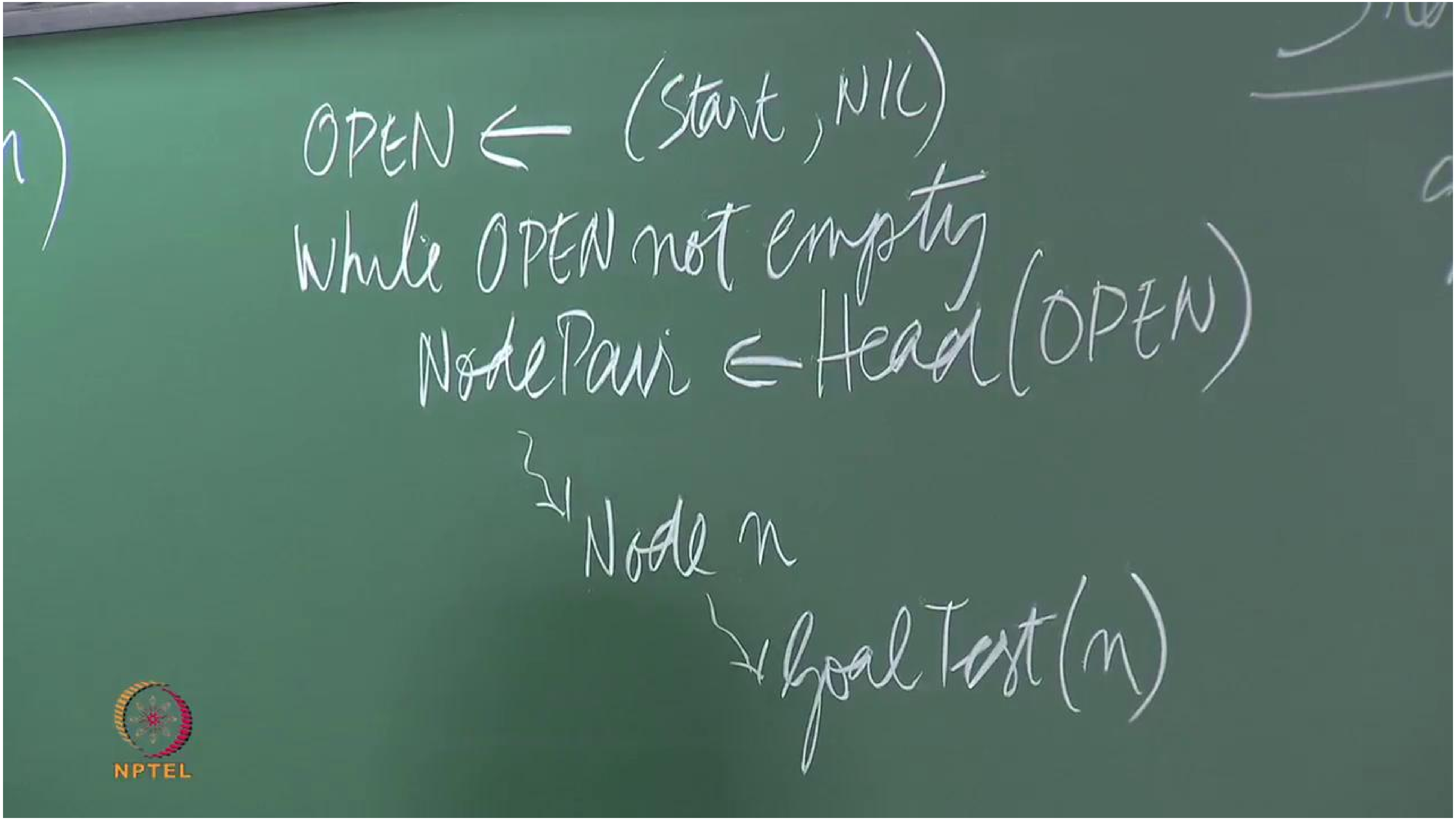


State Space Search → Search Tree

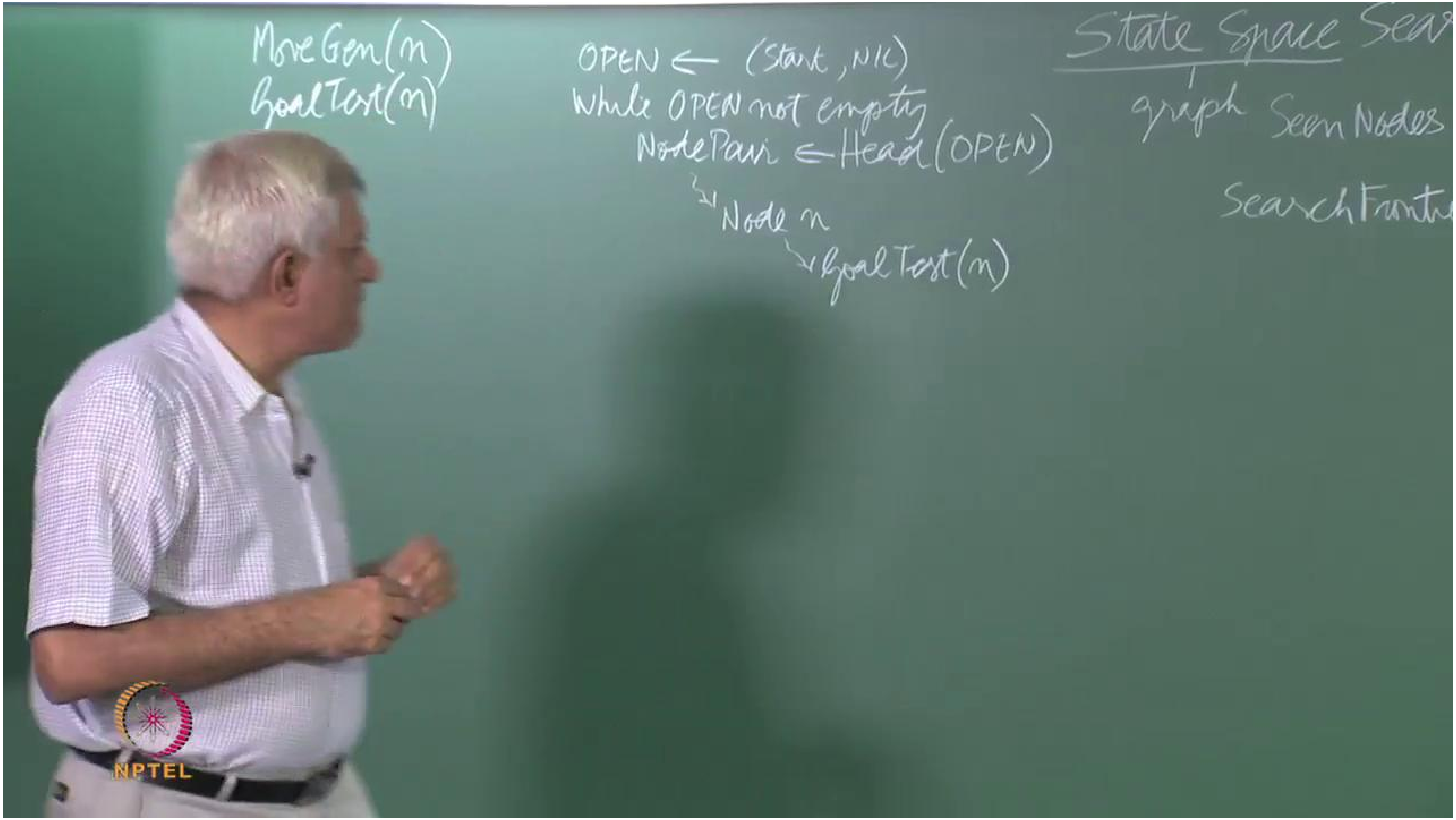
graph

Seen Nodes – Internal Nodes – CLOSED

Search Frontier → leaves = OPEN



```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
              ↳ GoalTest(n)
```



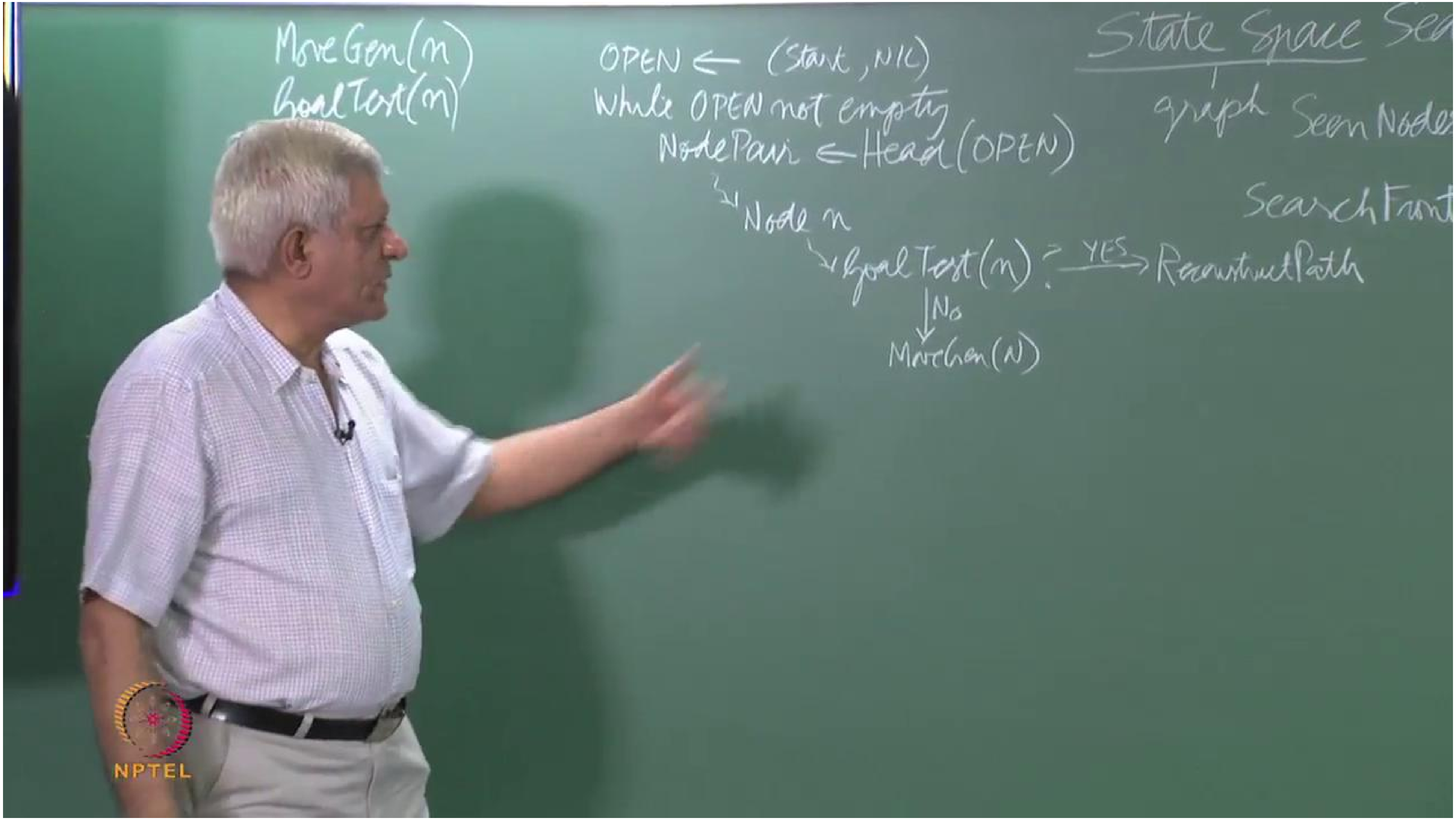
MoveGen(n)
GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n)
```

State Space Sear

graph Seen Nodes

Search Fronti



MoveGen(n)
GoalTest(n)

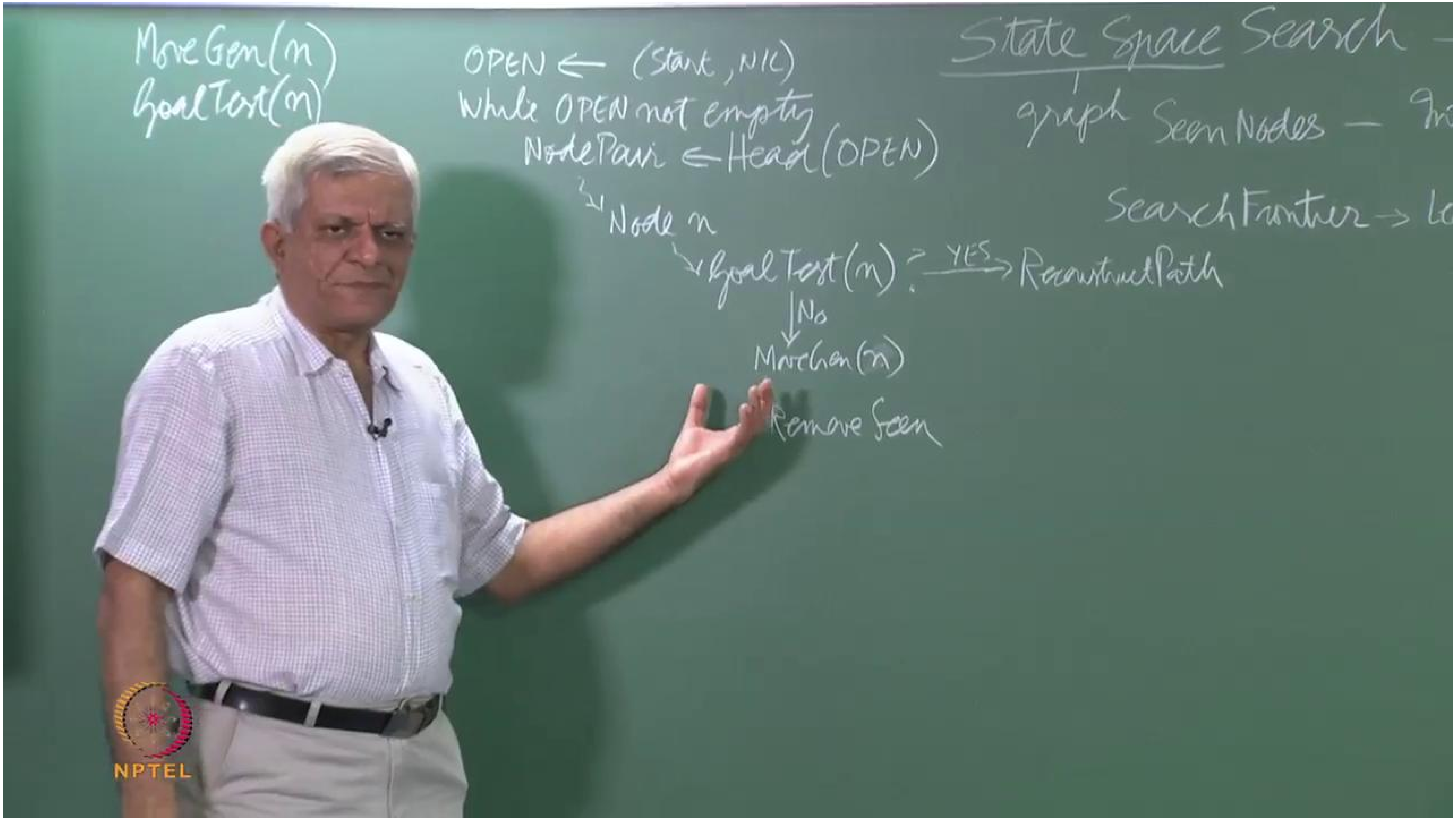
```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                  ↓ No
              MoveGen(N)
```

State Space Sea

graph

Seen Nodes

Search Front



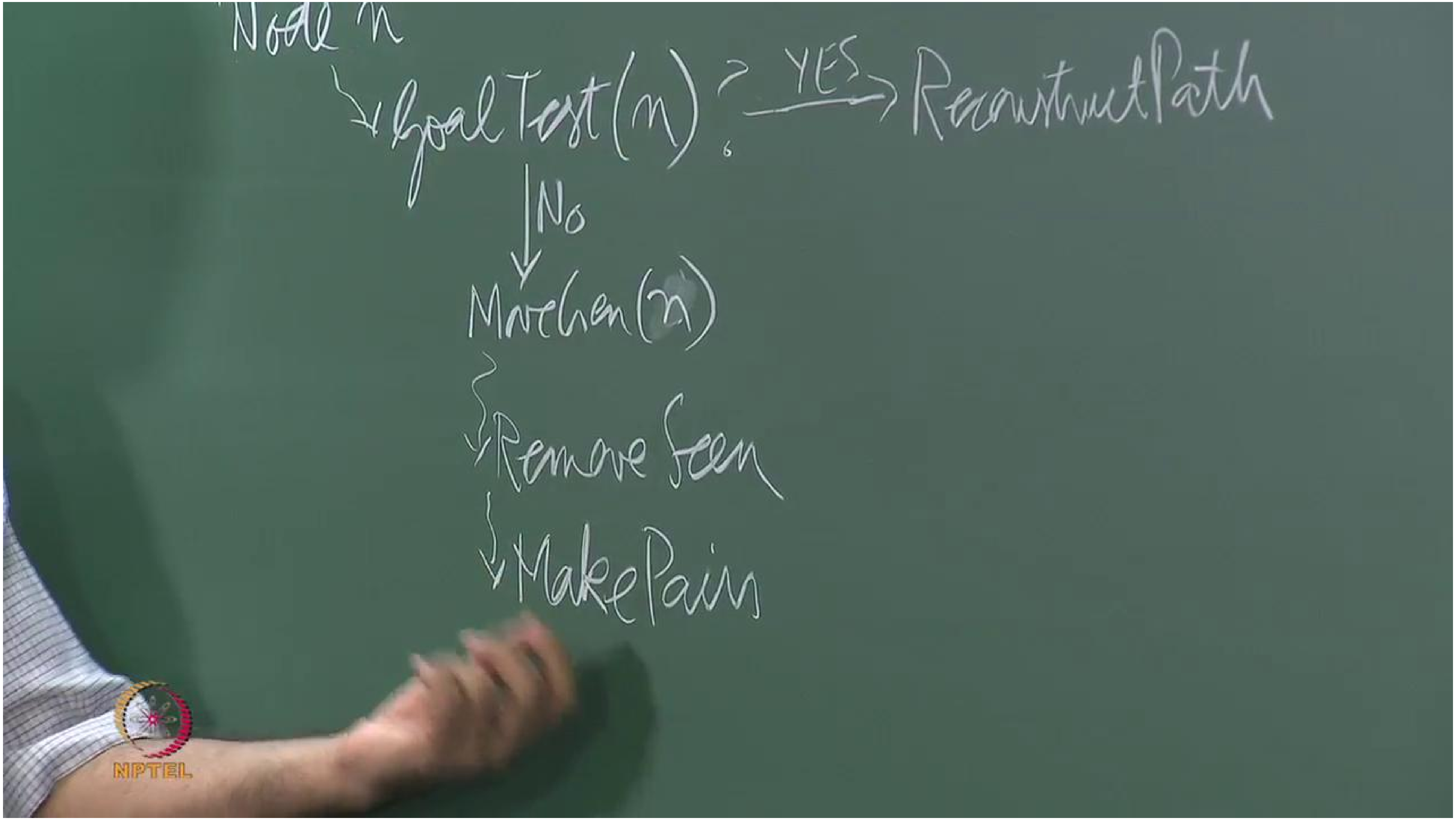
MoveGen(n)
GoalTest(n)

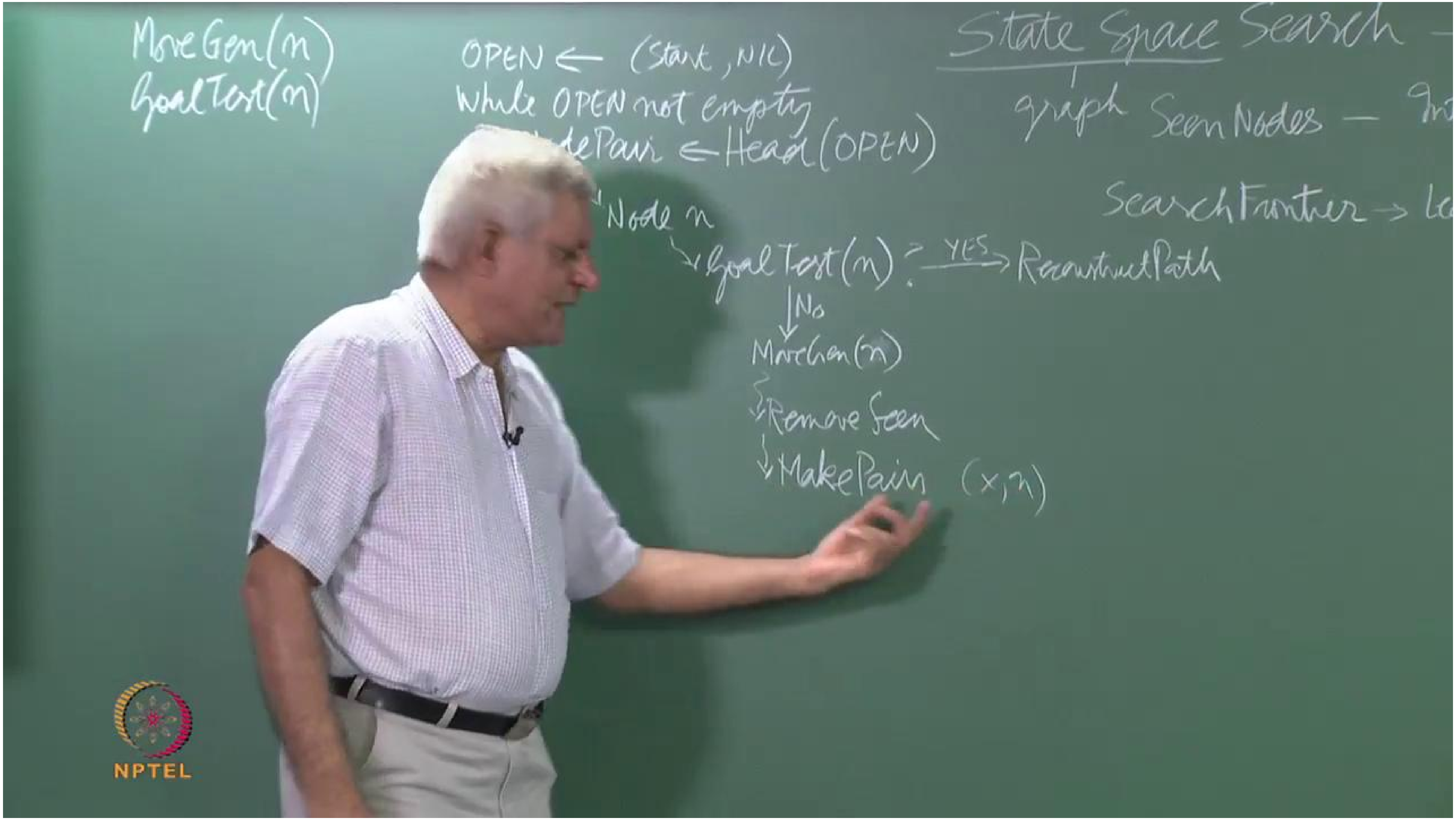
```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                ↓ No
                MoveGen(n)
                Remove Seen
```

State Space Search –

graph Seen Nodes – Tr

SearchFrontier → Lo


Node n
GoalTest(n)?
YES
ReconstructPath
No
MoveGen(n)
RemoveSeen
MakePairs



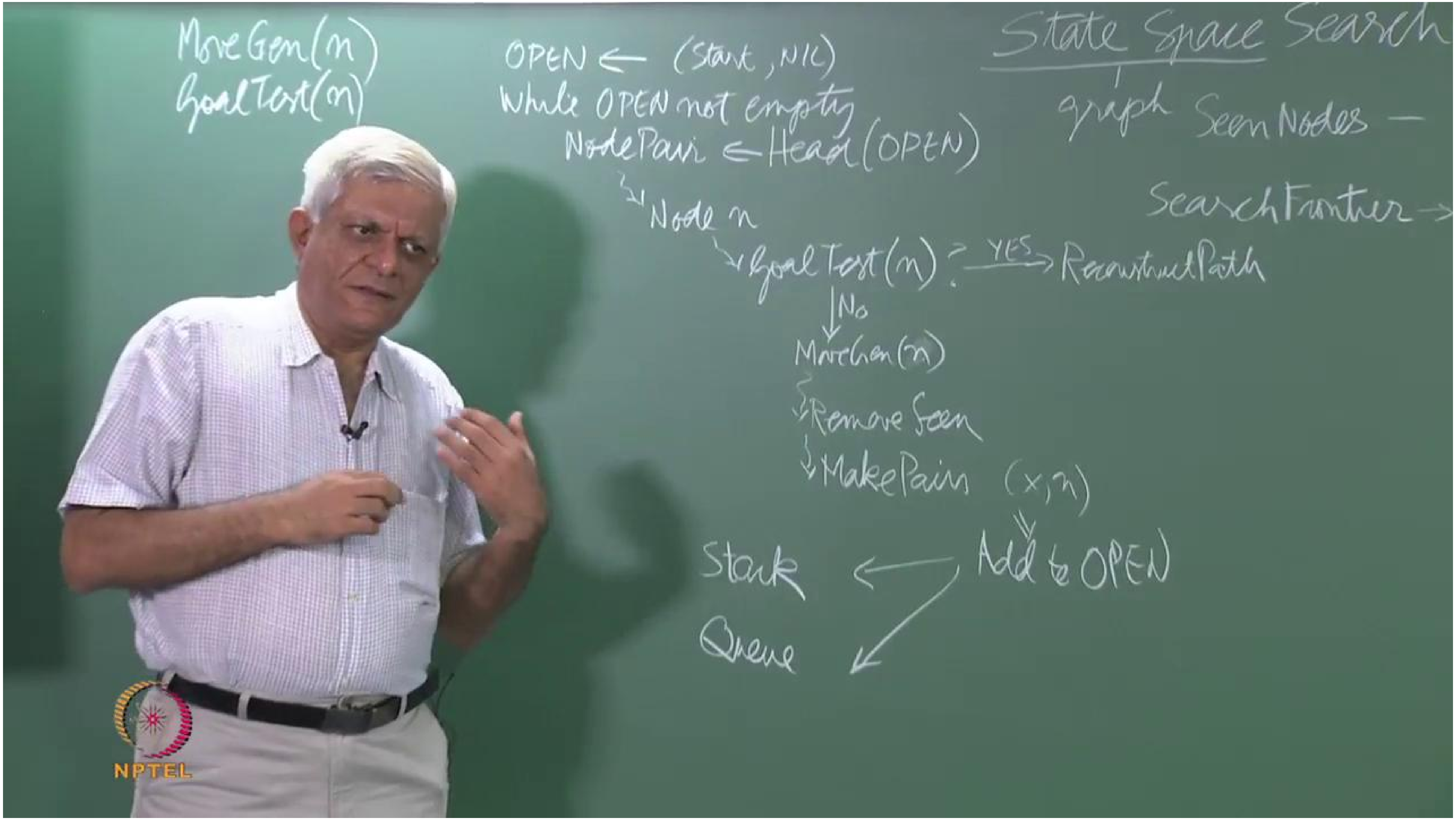
MoveGen(n)
GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    dePair ← Head(OPEN)
    Node n
        ↓ GoalTest(n) ? —YES→ ReconstructPath
            ↓ No
        MoveGen(n)
        ↓ Remove Seen
        ↓ MakePairs   (x, n)
```

State Space Search —

graph Seen Nodes —

SearchFrontier →



State Space Search

graph

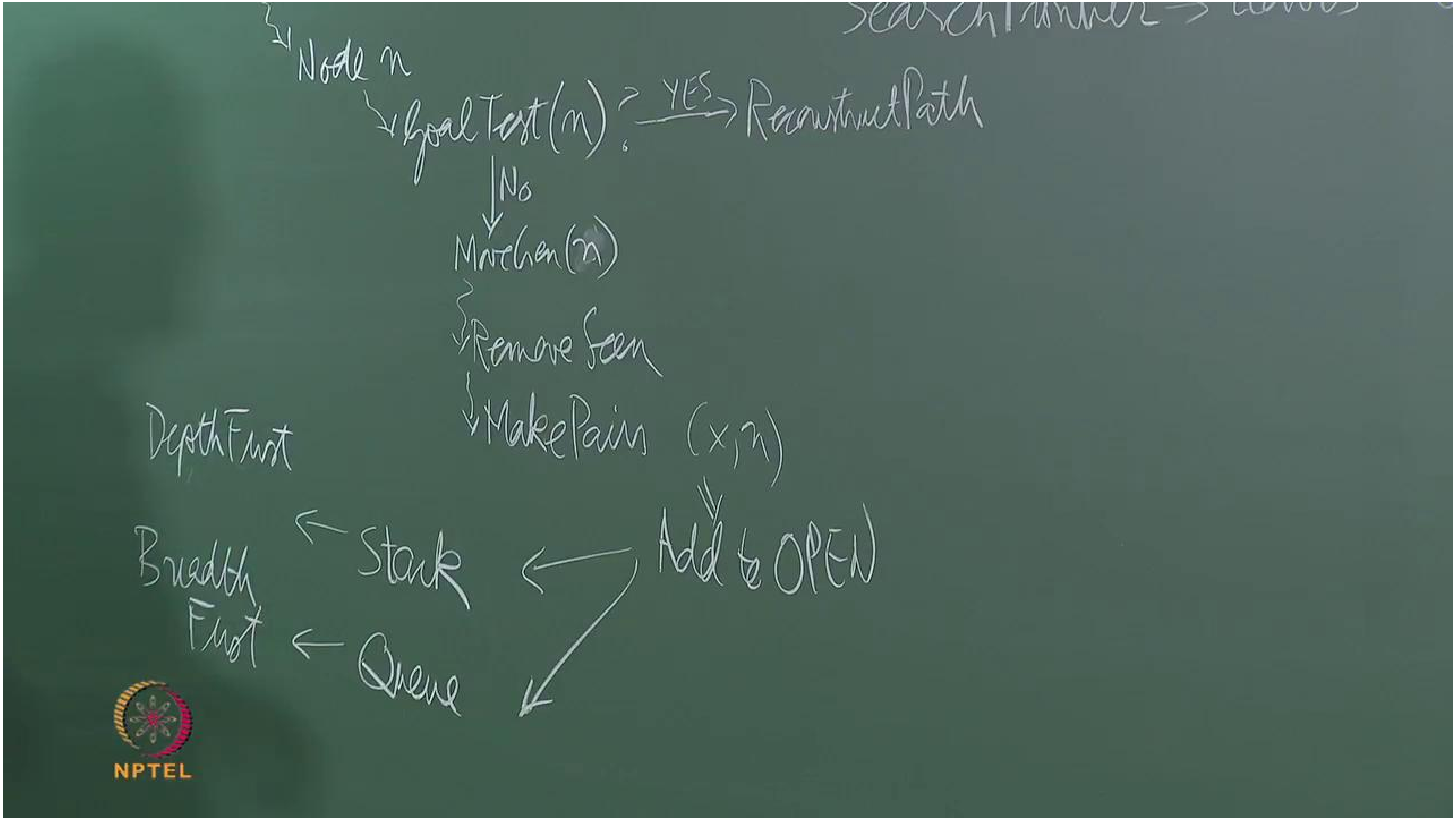
SeenNodes —

SearchFrontier →

MoveGen(n)

GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                ↓No
              MoveGen(n)
              ↳ Remove Seen
              ↳ MakePairs (x, n)
                              ⇓
    Stack ←          Add to OPEN
    Queue ↙
```

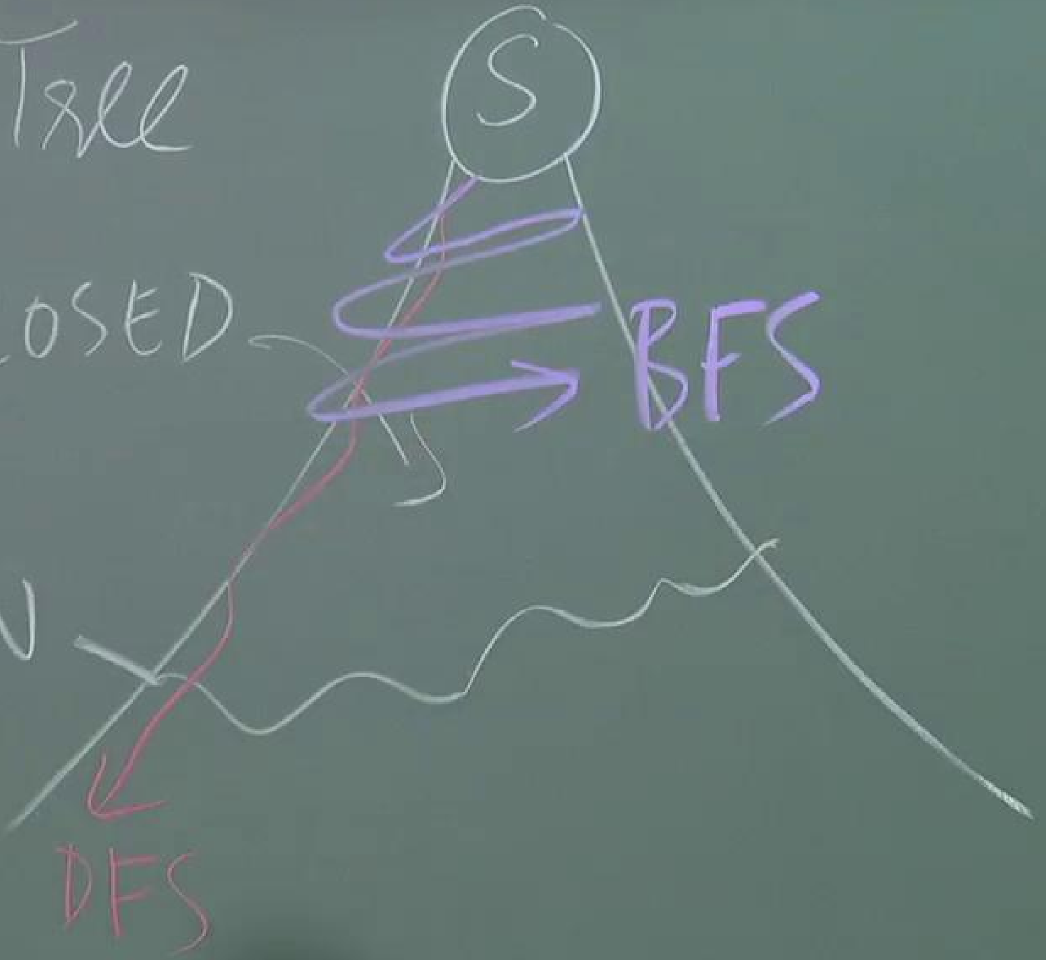

SearchProblem →
Node n
GoalTest(n) ?
YES
ReconstructPath
No
MoveGen(n)
RemoveSeen
MakePairs (x,n)
Add to OPEN
Stack
Queue
DepthFirst
Breadth First
NPTEL

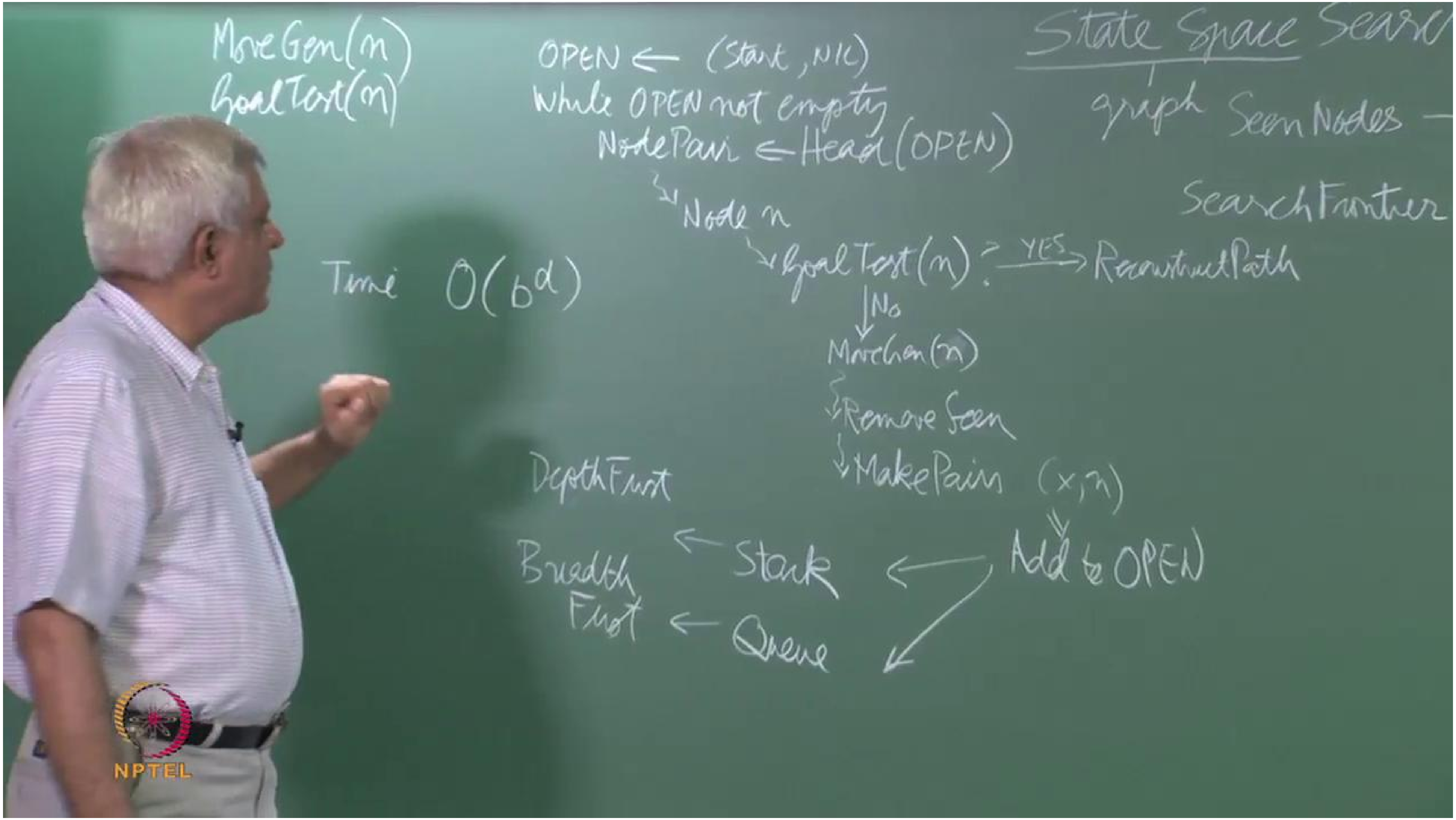
Search Tree

— CLOSED

des

= OPEN





State Space Search

graph

SeenNodes —

SearchFrontier

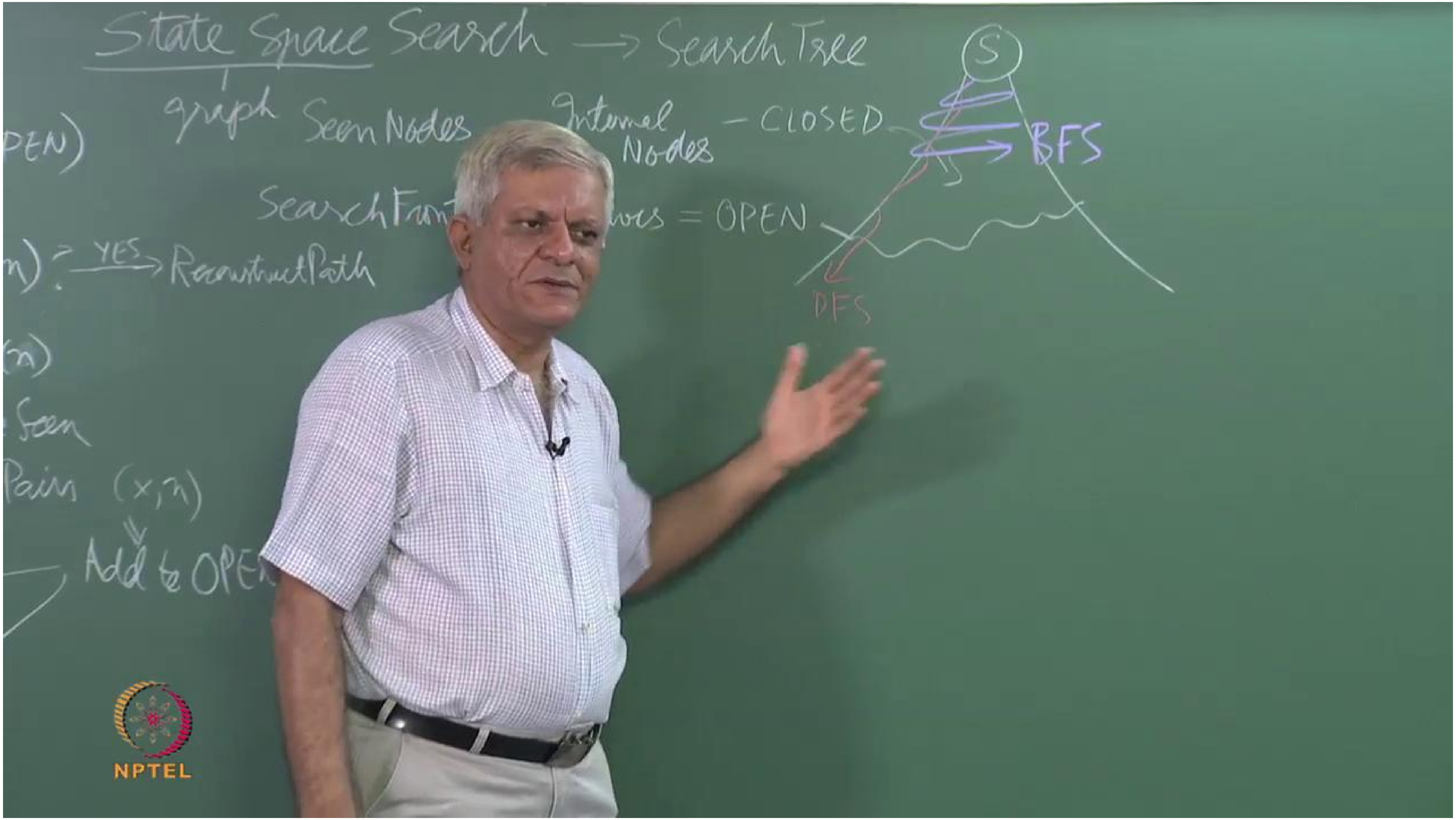
MoveGen(n)
GoalTest(n)

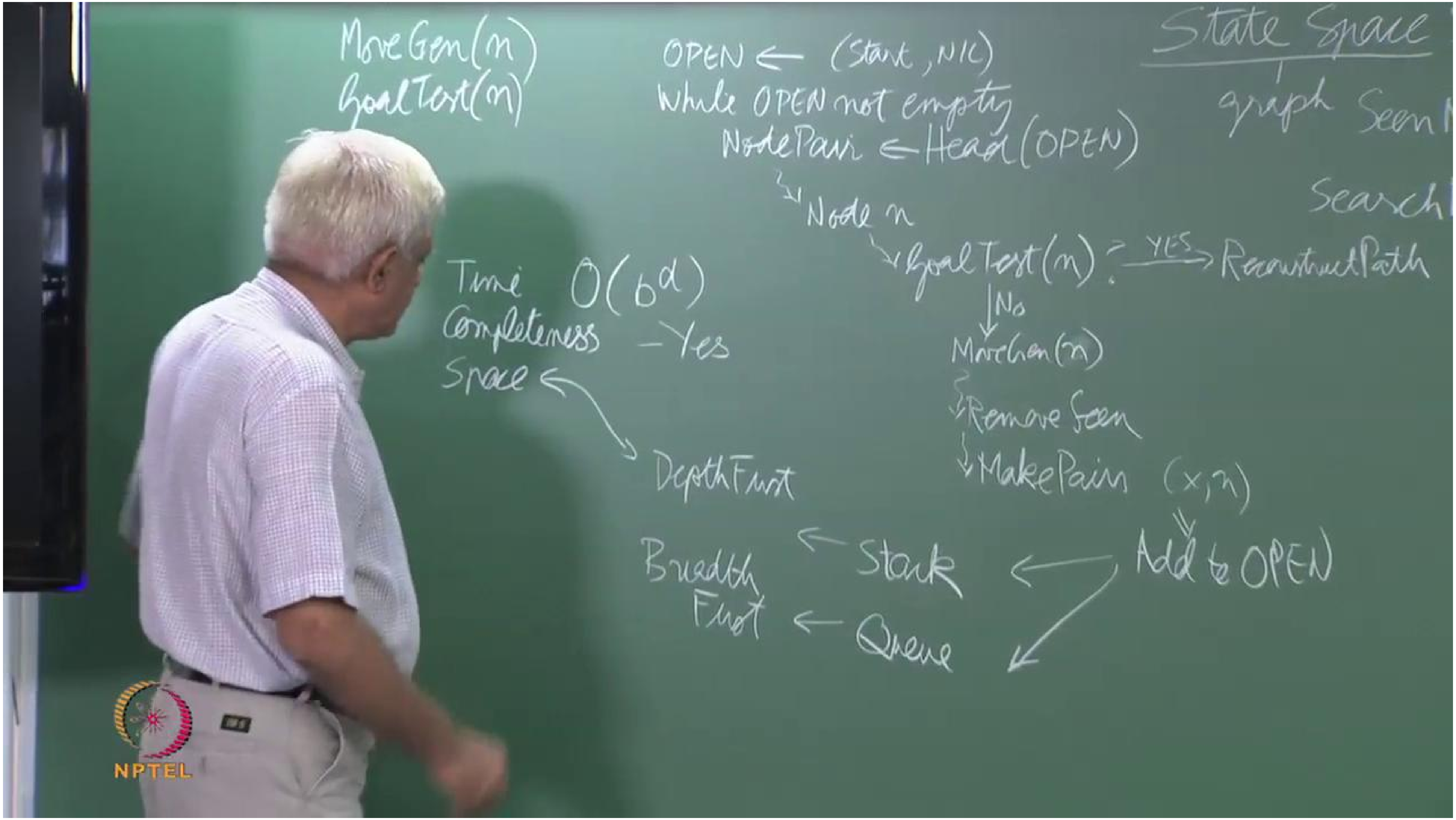
```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                ↓ No
              MoveGen(n)
                ↳ Remove Seen
                ↳ MakePairs (x, n)
                        ⇓
                      Add to OPEN
```

Time $O(b^d)$

DepthFirst ← Stack ← Add to OPEN

Breadth First ← Queue ← Add to OPEN


State Space Search → Search Tree
graph
Seen Nodes
Internal Nodes – CLOSED
Search Fron... ...ves = OPEN
PEN)
n) ? YES → ReconstructPath
(n)
Seen
Pair (x,n)
Add to OPEN
S
BFS
DFS
NPTEL



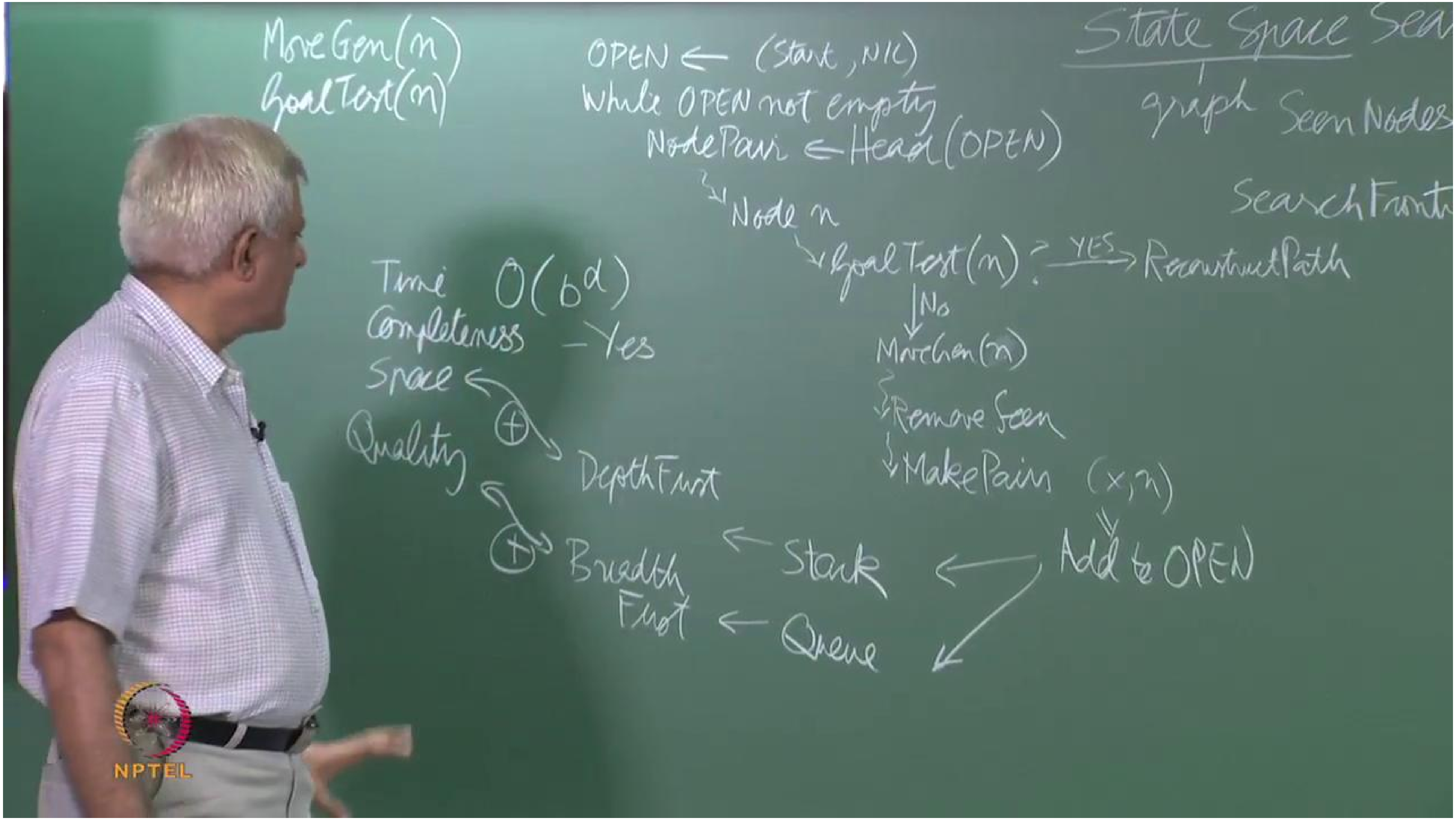
State Space
graph Seen
Search

MoveGen(n)
GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                ↓ No
              MoveGen(n)
                ↳ Remove Seen
                ↳ MakePairs (x, n)
                    ↳ Add to OPEN
```

Time $O(b^d)$
Completeness – Yes
Space ← → Depth First

Depth First ← Stack ← Add to OPEN
Breadth First ← Queue ← Add to OPEN



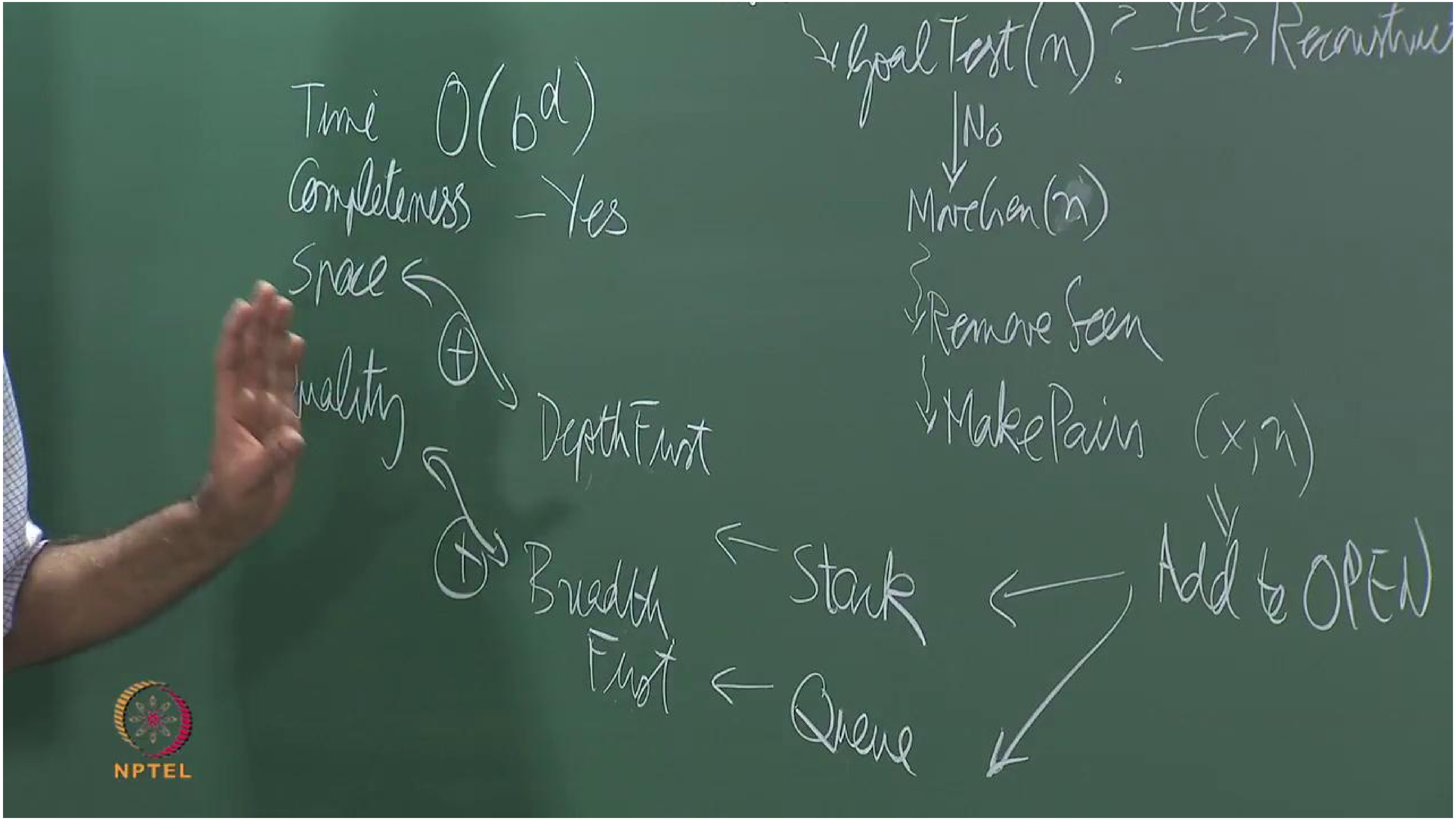
MoveGen(n)
GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                ↓ No
              MoveGen(n)
                ↳ Remove Seen
                ↳ MakePairs (x, n)
                              ⇓
                          Add to OPEN
```

State Space Sea
graph
Seen Nodes
SearchFronti

Time $O(b^d)$
Completeness — Yes
Space
Quality

⊕ DepthFirst ← Stack ← Add to OPEN
⊕ Breadth First ← Queue ← Add to OPEN



Goal Test(n) ? $\xrightarrow{\text{Yes}}$ Reconstruct

No

MoveGen(n)

Remove Seen

MakePairs (x, n)

Add to OPEN

Stack

Queue

Depth First

Breadth First

Time $O(b^d)$

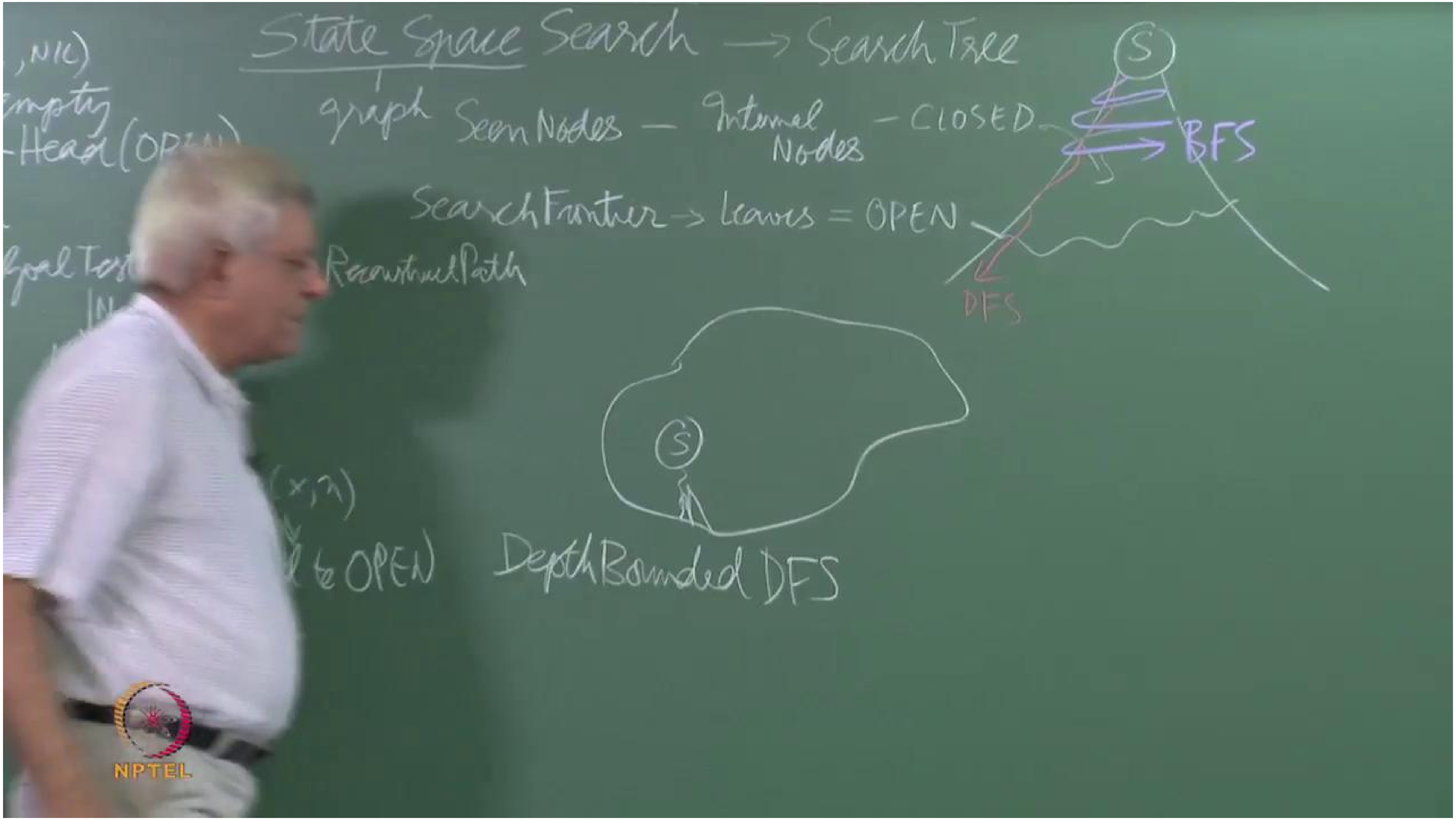
Completeness - Yes

Space

Quality

$\oplus$

$\otimes$



, NIL)

empty

Head(OPEN)

goal Test

|N

(x, n)

l to OPEN

State Space Search → Search Tree

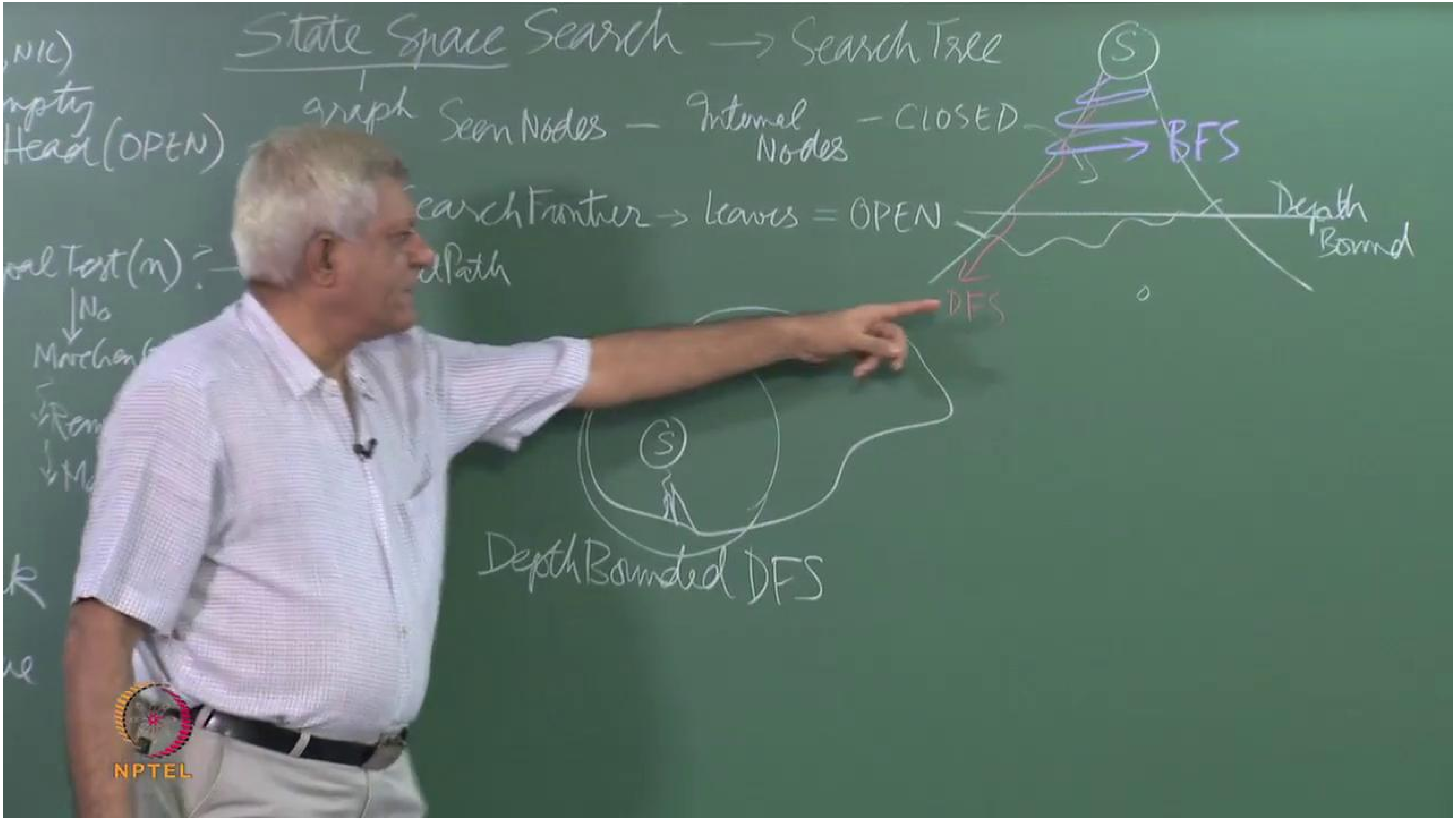
graph

Seen Nodes — Internal Nodes - CLOSED

Search Further → leaves = OPEN

ReconstructPath

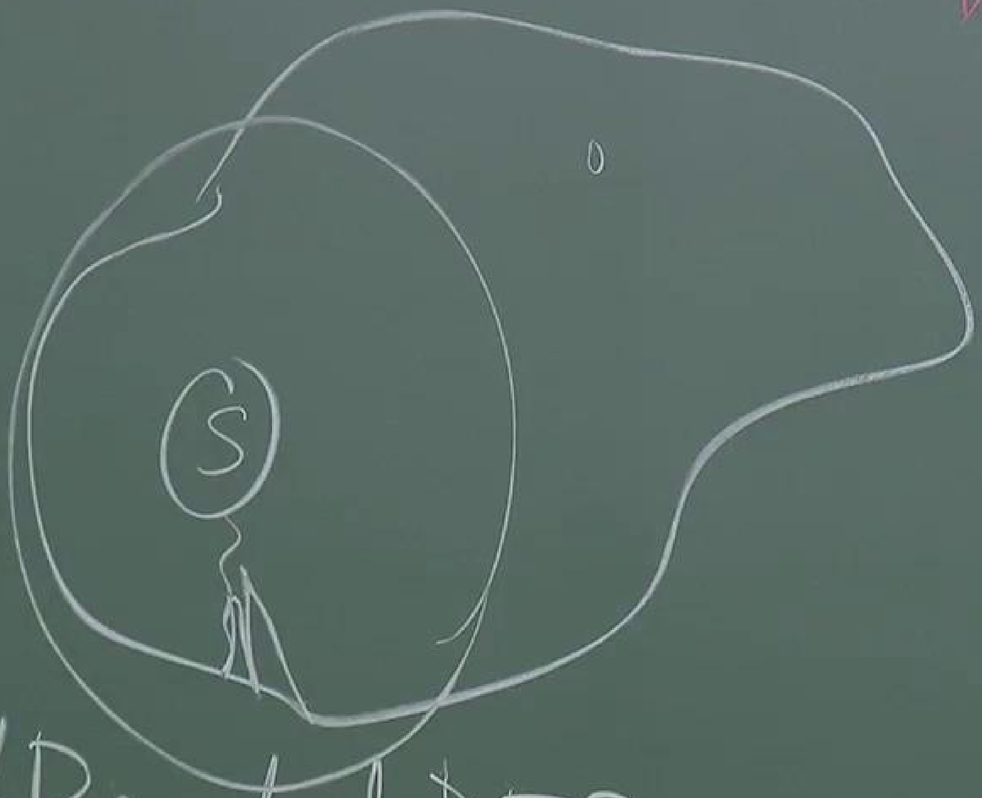
Depth Bounded DFS



State Space Search → Search Tree
graph
Seen Nodes — Internal Nodes — CLOSED
Search Frontier → Leaves = OPEN
dPath
,NIC)
mpty
Head(OPEN)
oal Test(n) ?
No
Movechen
Rem
Mo
k
ue
S
BFS
Depth Bound
DFS
S
Depth Bounded DFS
NPTEL

tPath

DFS



Depth Bounded DFS

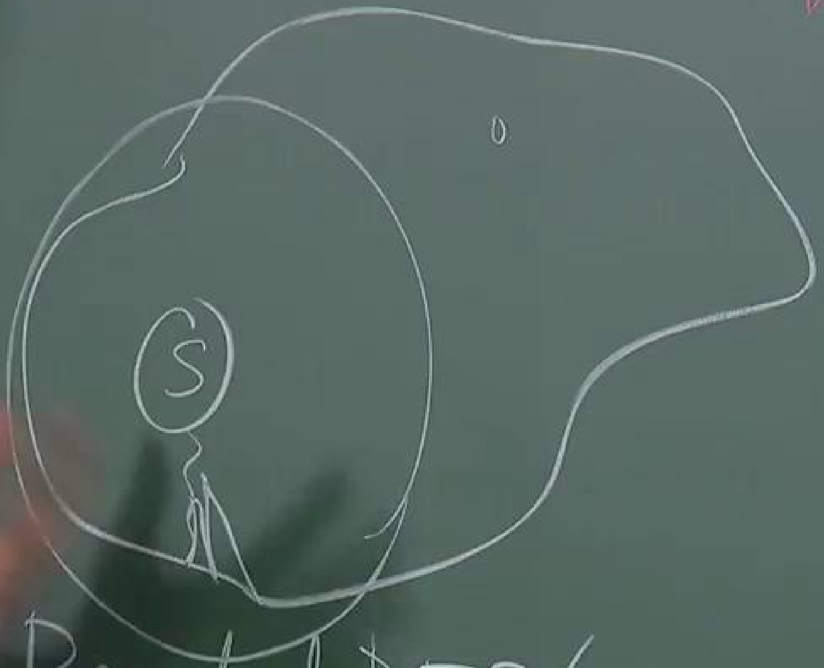
DFS

Depth Bounded DFS

DepthBound ← 0

[illegible]

DFS

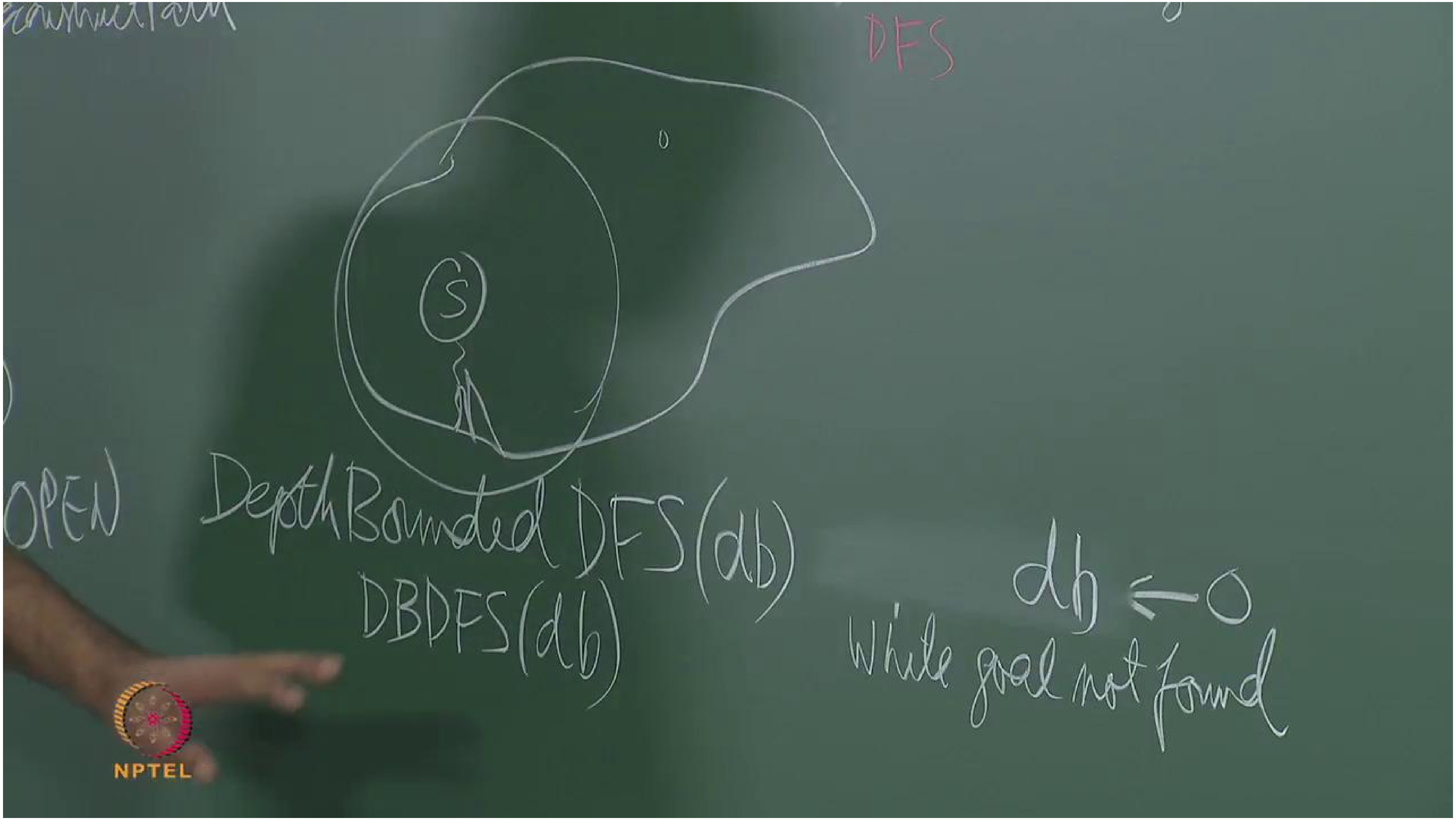


Depth Bounded DFS(db)

DBDFS(db)

db ← 0

while goal not found

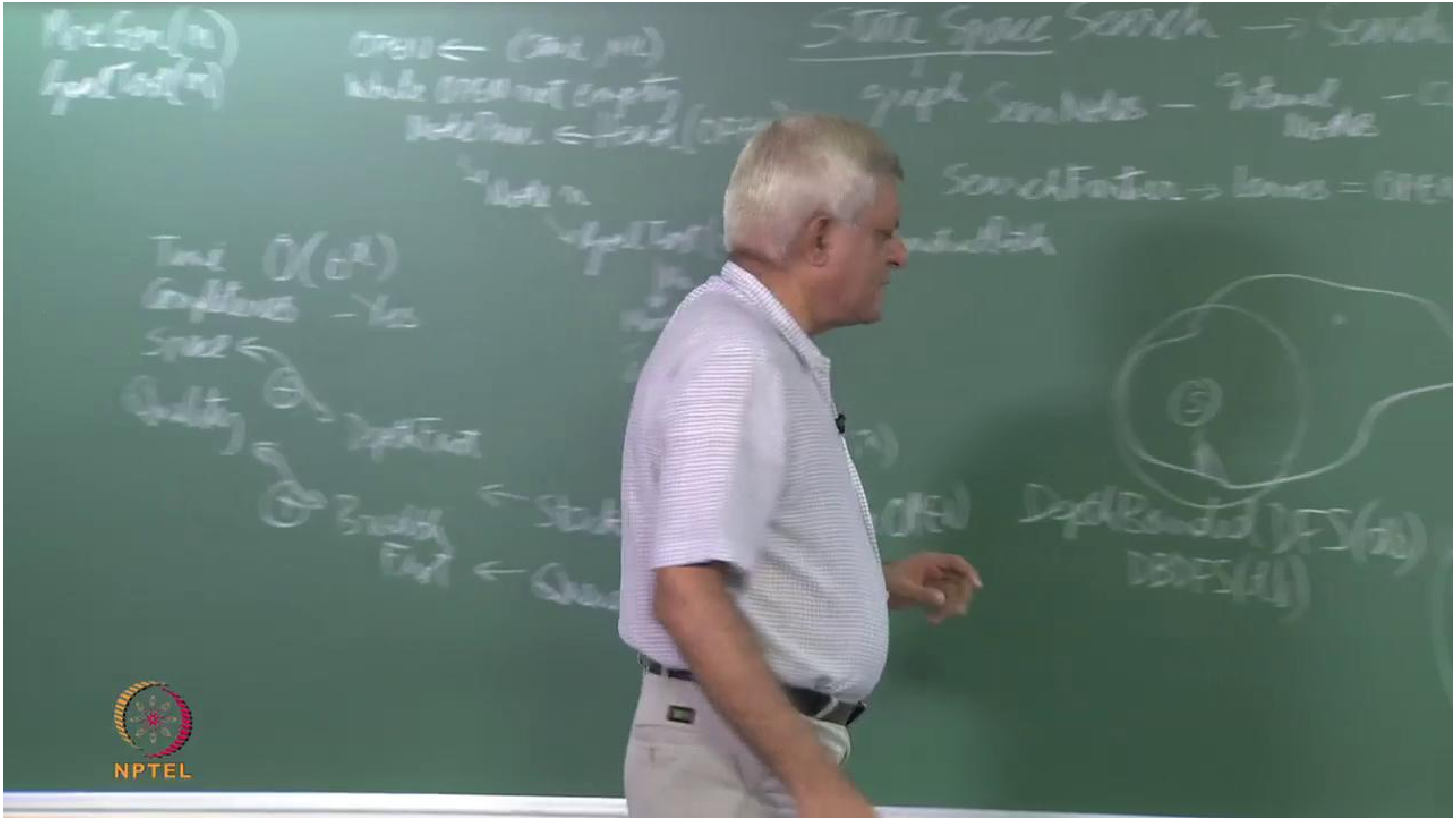


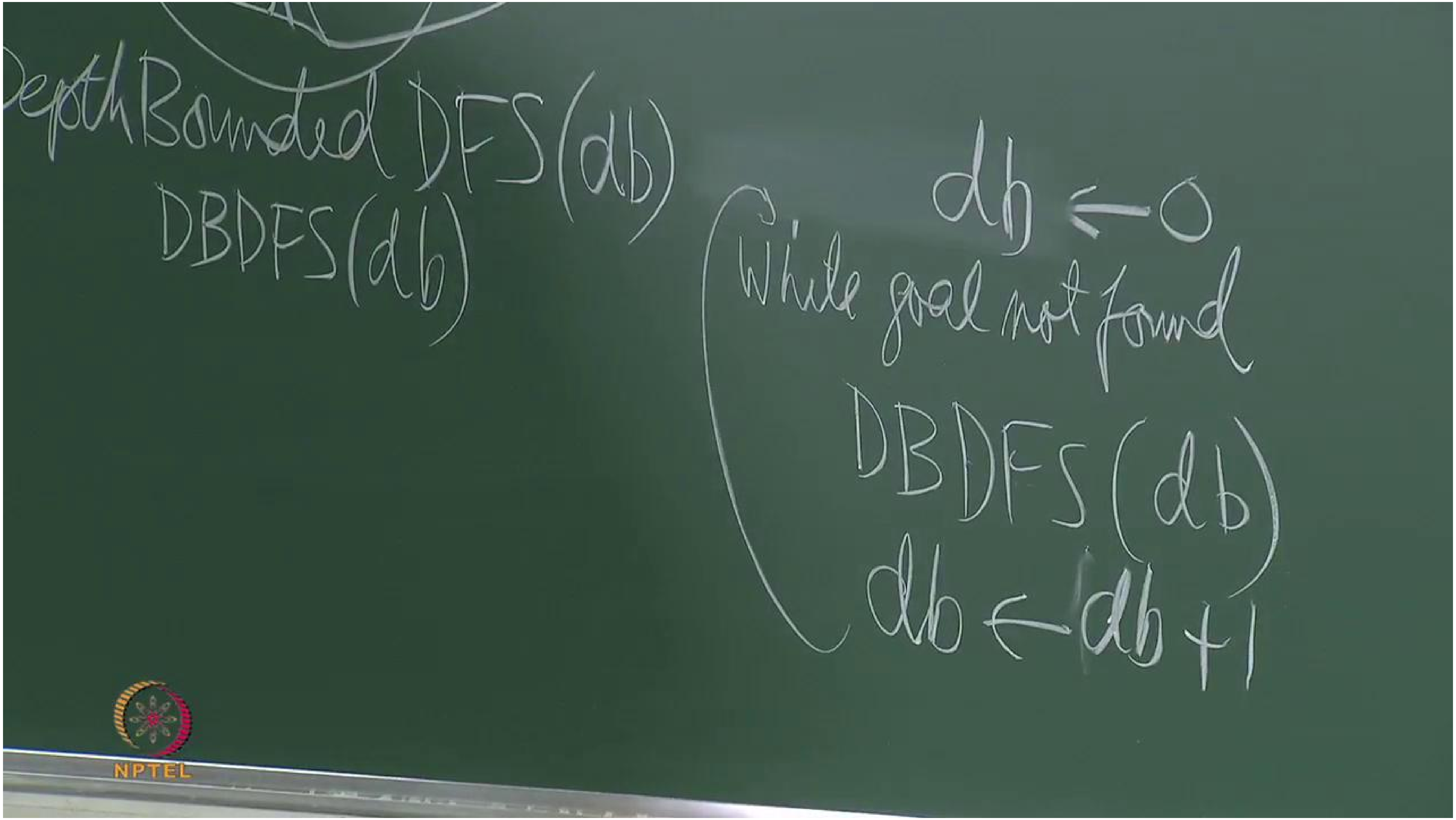
Depth Bounded DFS(db)

DBDFS(db)

db ← 0

While goal not found

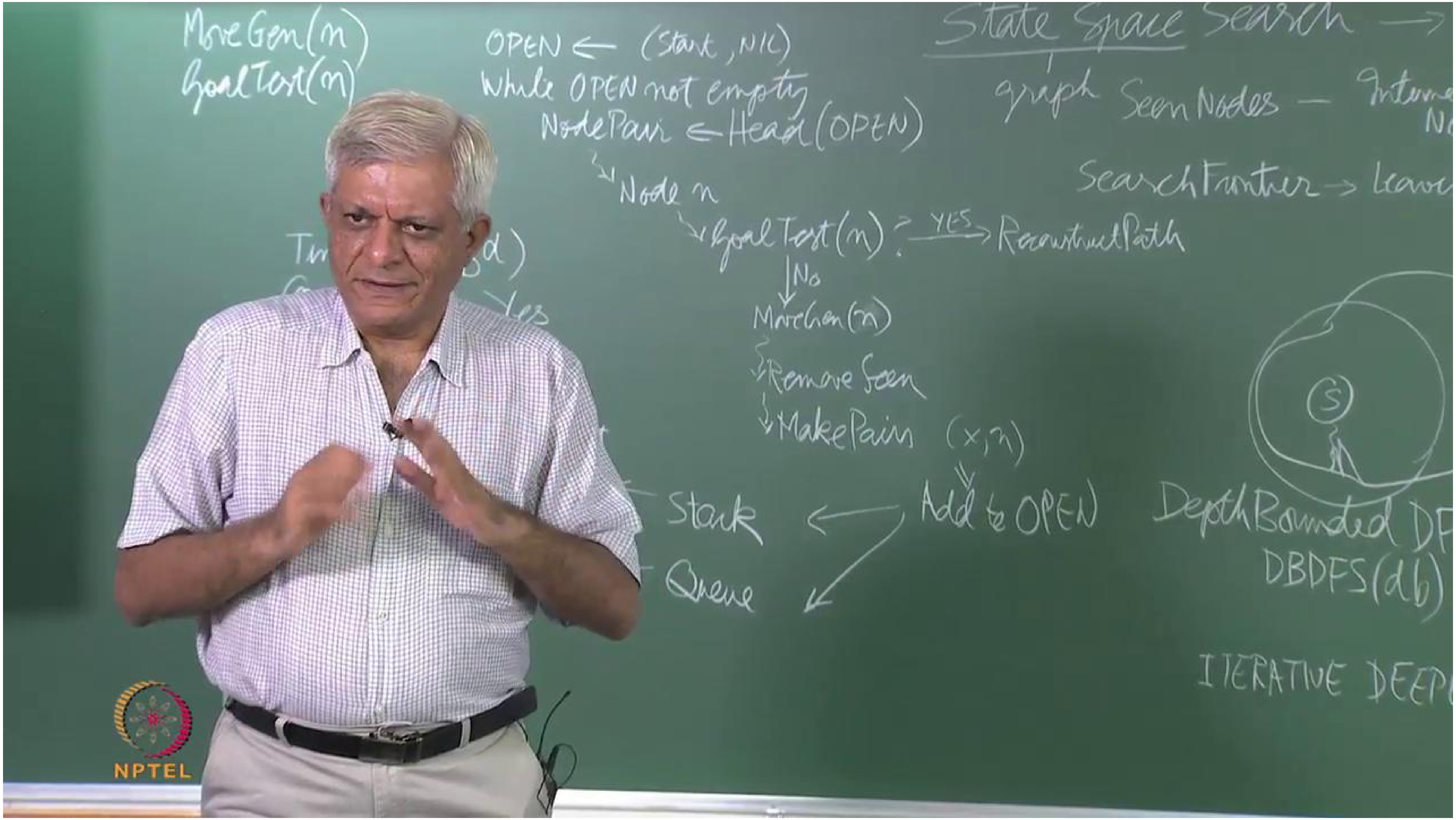

MoveGen(m)
GoalTest(m)
OPEN ← (start, nil)
While OPEN not empty
NodePair ← Head(OPEN)
Node n
GoalTest
State Space Search → Search
graph
Search Nodes — Internal Nodes
SearchFrontier → leaves = OPEN
Time O(b^d)
Completeness — Yes
Space
Quality
DepthFirst
Breadth First
← Stack
← Queue
(OPEN)
S
DepthBounded DFS(20)
DBDFS(d)



Depth Bounded DFS(db)

DBDFS(db)

```
db ← 0
While goal not found
    DBDFS(db)
    db ← db + 1
```



MoveGen(n)
GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                ↓No
              MoveGen(n)
                ↳ Remove Seen
                ↳ MakePairs (x, n)
                          Add to OPEN
```

Stack ←

Queue

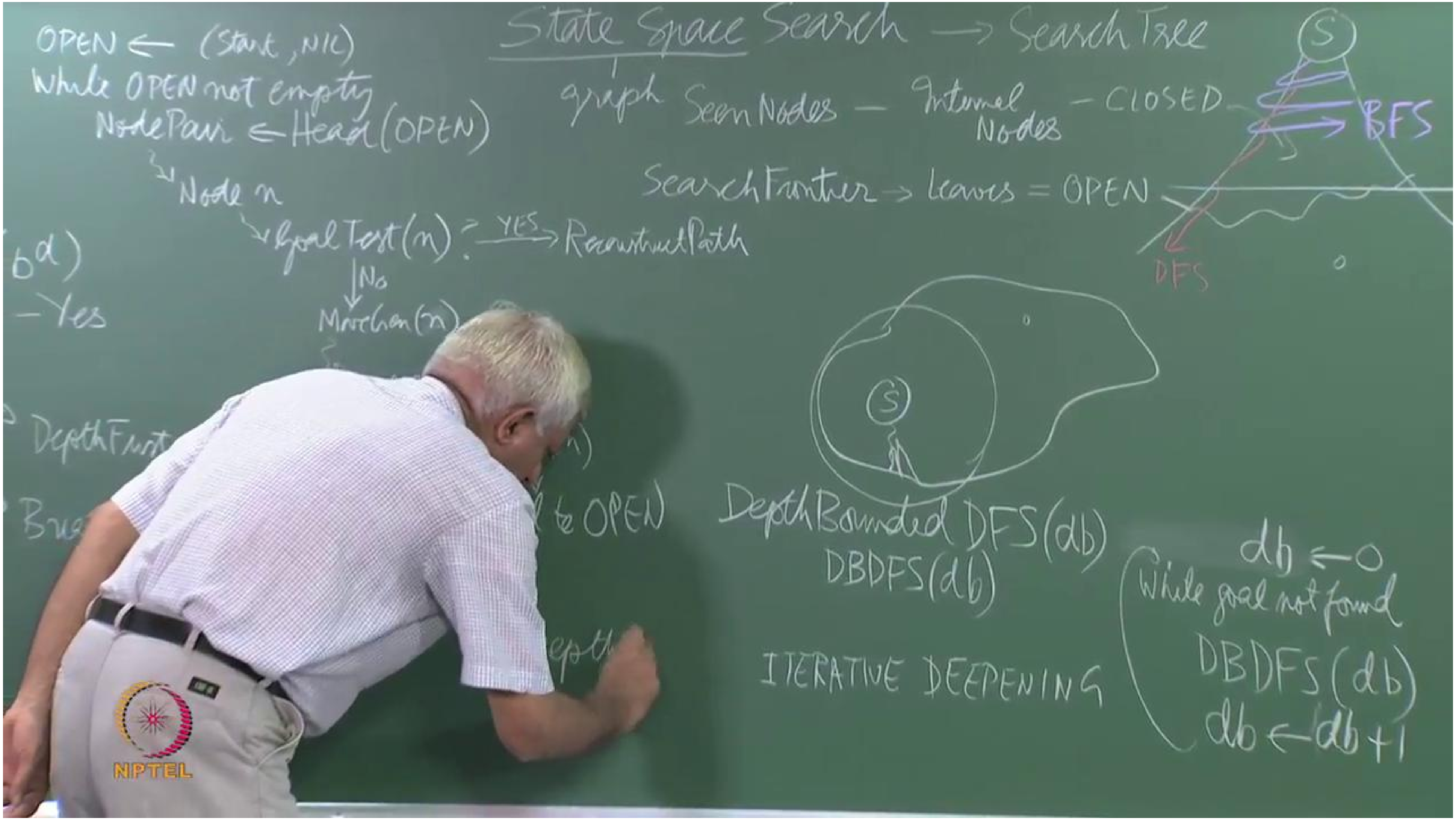
State Space Search →

graph   Seen Nodes — Interna
No

SearchFrontier → Leave

DepthBounded DF
DBDFS(db)

ITERATIVE DEEP



State Space Search → Search Tree

graph   Seen Nodes — Internal Nodes — CLOSED

SearchFrontier → Leaves = OPEN

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                  ↓No
              MoveGen(n)
              n)
              l to OPEN
              epth
```

(b^d)

— Yes

DepthFirst

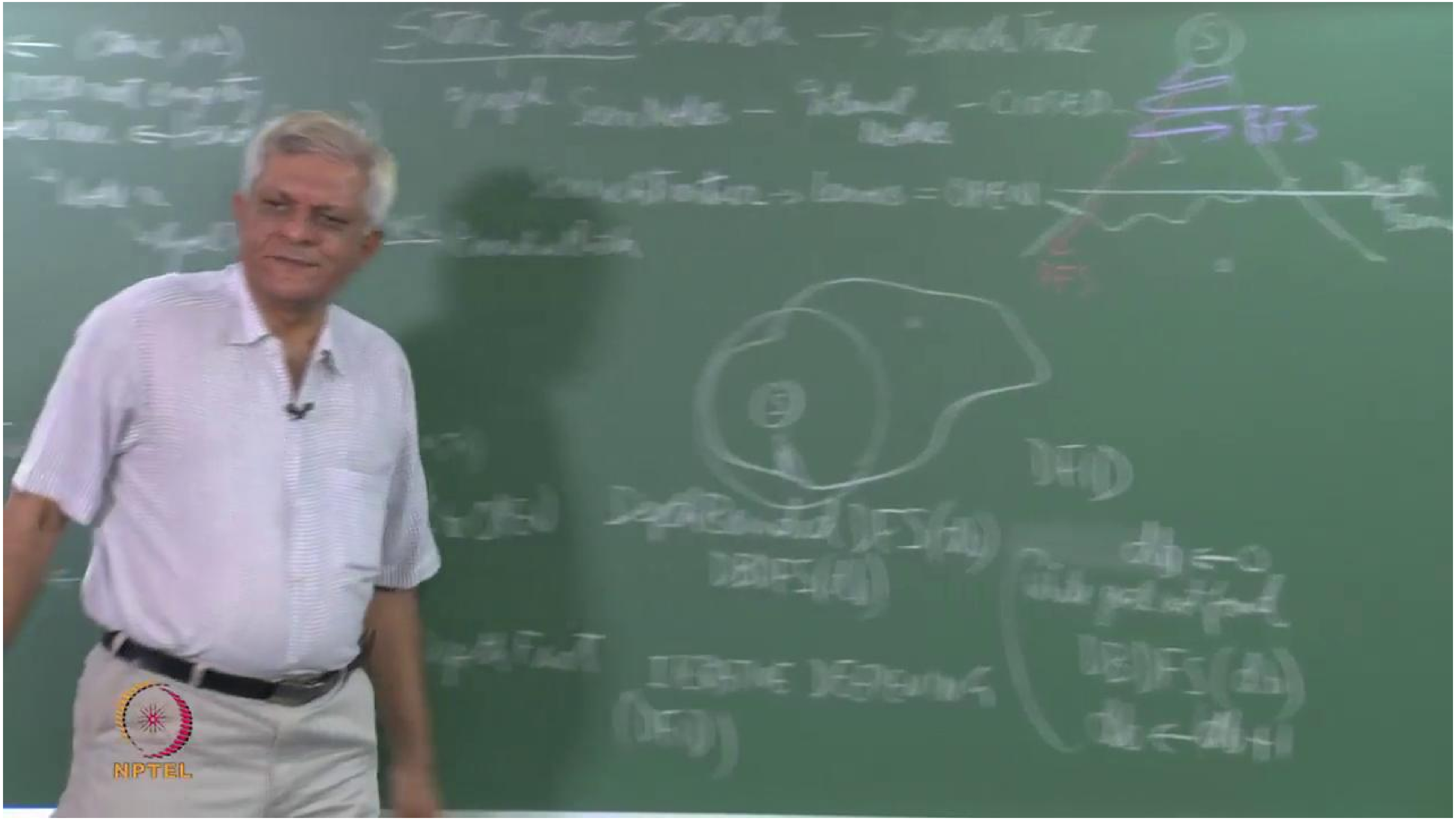
Brea

Depth Bounded DFS(db)

DBDFS(db)

ITERATIVE DEEPENING

```
db ← 0
While goal not found
    DBDFS(db)
    db ← db + 1
```


State Space Search → Search Tree
graph Search Nodes – Visited nodes – CLOSED
BFS
Search Frontier → leaves = OPEN
Depth
DFS
DFID
Depth Bounded DFS (db)
DBDFS(db)
db ← 0
while goal not found
DBDFS(db)
db ← db+1
Iterative Deepening
(DFID)
NPTEL

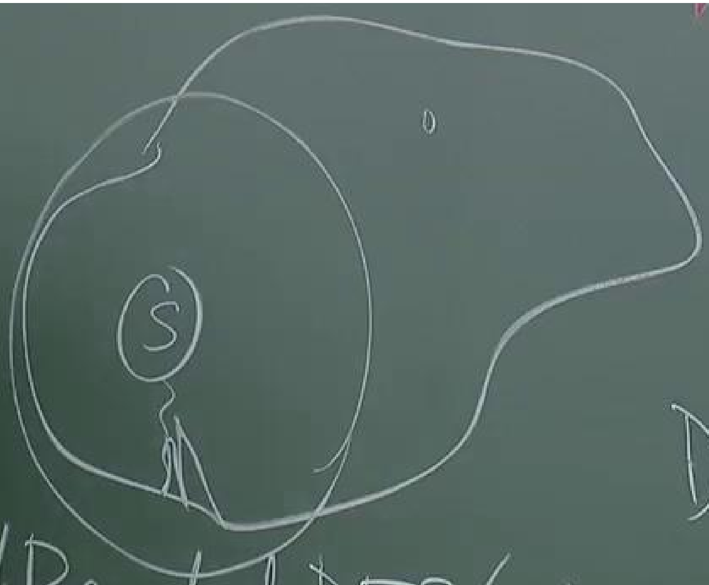
en(n)

ave seen

kePair (x,n)

⇓

Add to OPEN



Depth Bounded DFS(db)

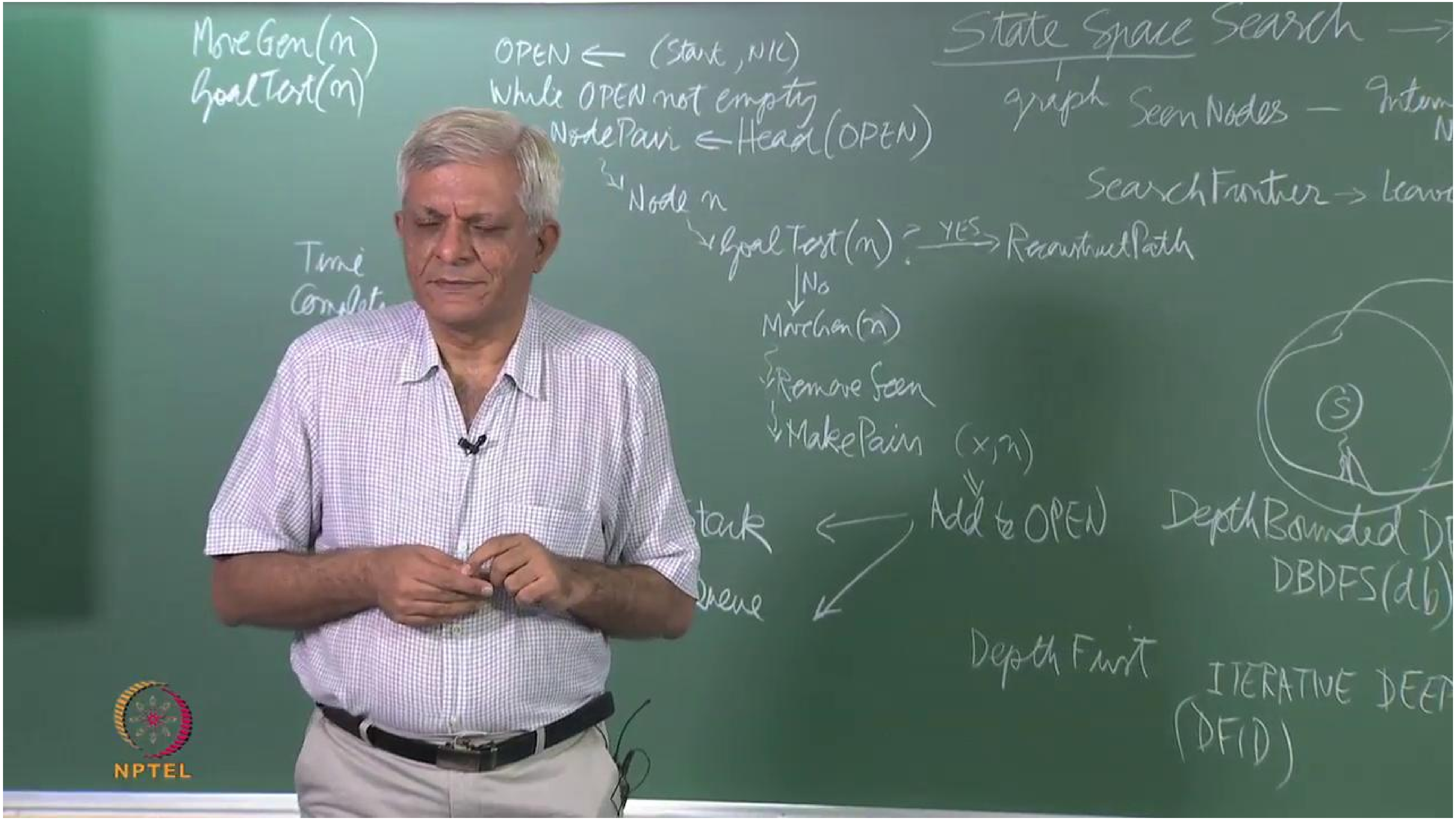
DBDFS(db)

Depth First

ITERATIVE DEEPENING (DFID)

DFID

```
db ← 0
While goal not found
    DBDFS(db)
    db ← db + 1
```



MoveGen(n)
GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        ↳ Node n
            ↳ GoalTest(n) ? —YES→ ReconstructPath
                ↓ No
              MoveGen(n)
                ↳ Remove Seen
                ↳ MakePairs (x, n)
                                ⇓
                           Add to OPEN
```

Time
Complete

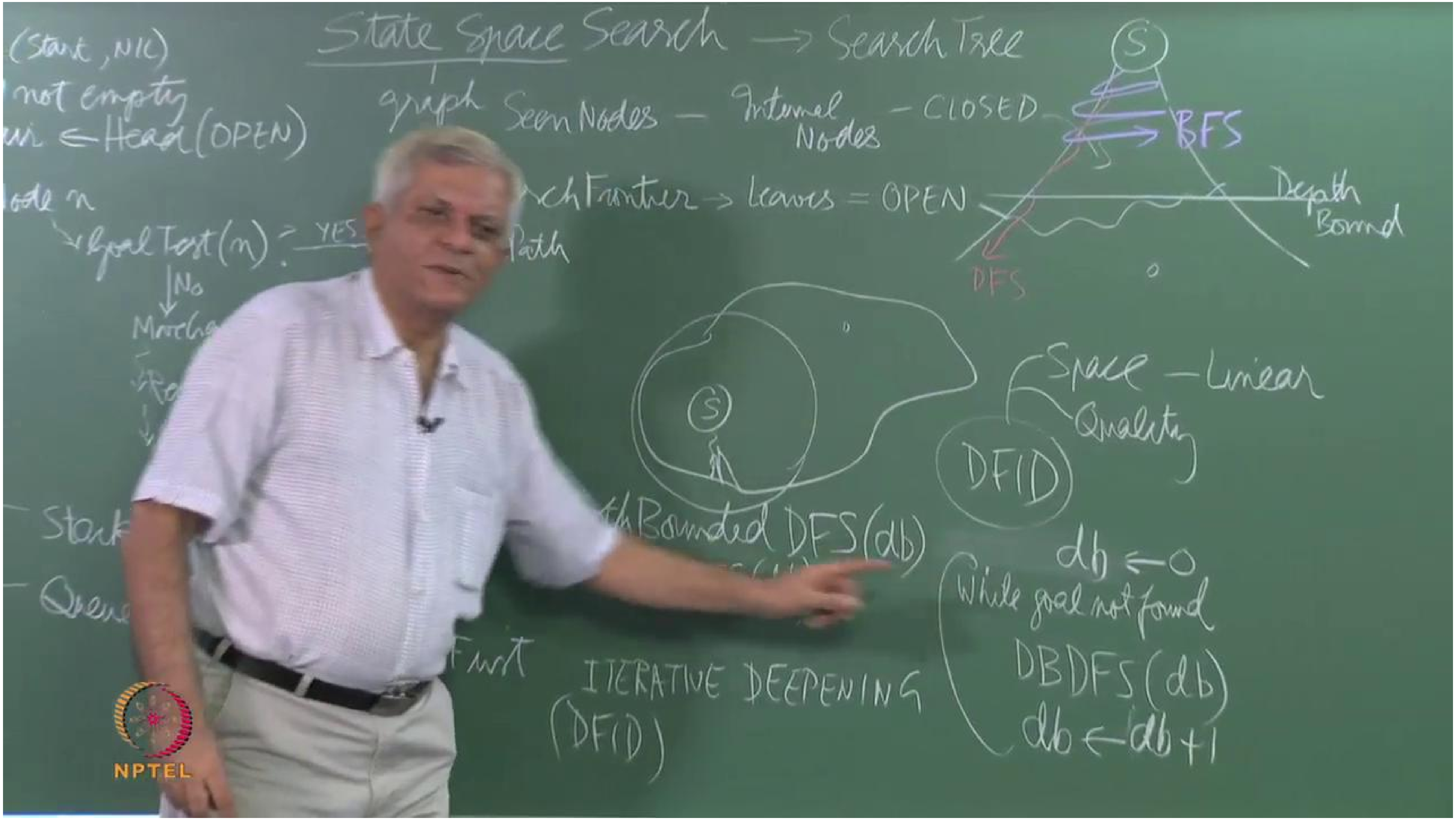
tack
ueue

Depth First

State Space Search →
graph
Seen Nodes — Intern
N
SearchFrontier → Leave

DepthBounded DF
DBDFS(db)

ITERATIVE DEEP
(DFID)



(Start, NIL)

not empty

← Head(OPEN)

node n

GoalTest(n) = YES — Path

↓No

MoveGen

Stack

Queue

First

State Space Search → Search Tree

graph

Seen Nodes — Internal Nodes — CLOSED

Search Frontier → leaves = OPEN

Space — Linear

Quality

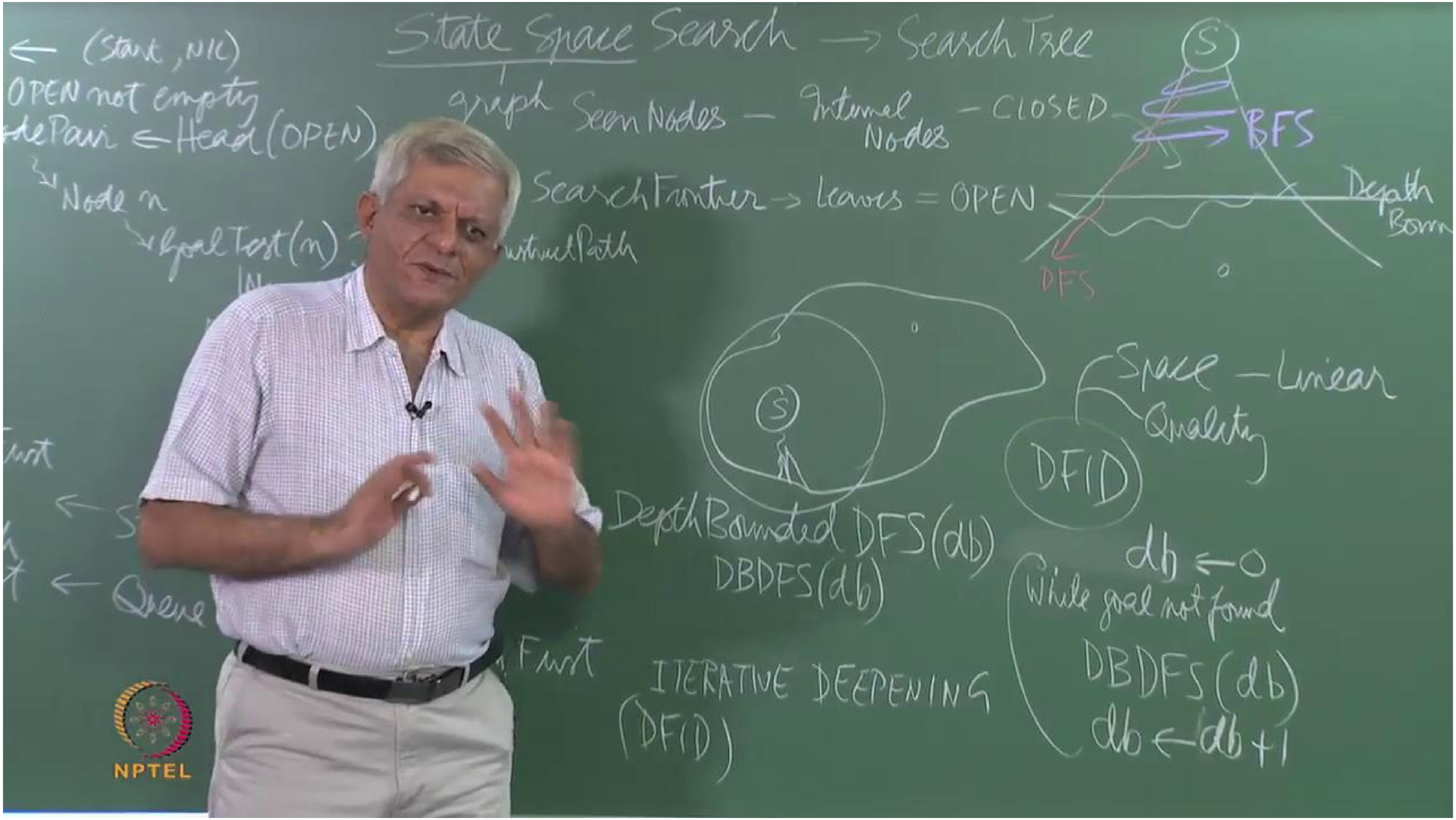
DFID

Depth Bounded DFS(db)

ITERATIVE DEEPENING (DFID)

```
db ← 0
While goal not found
    DBDFS(db)
    db ← db + 1
```

NPTEL



State Space Search → Search Tree

graph

Seen Nodes – Internal Nodes – CLOSED

Search Frontier → Leaves = OPEN

← (Start, NIL)

OPEN not empty

dePair ← Head(OPEN)

Node n

goalTest(n)

IN

structPath

First

← S

← Queue

First

Space – Linear

Quality

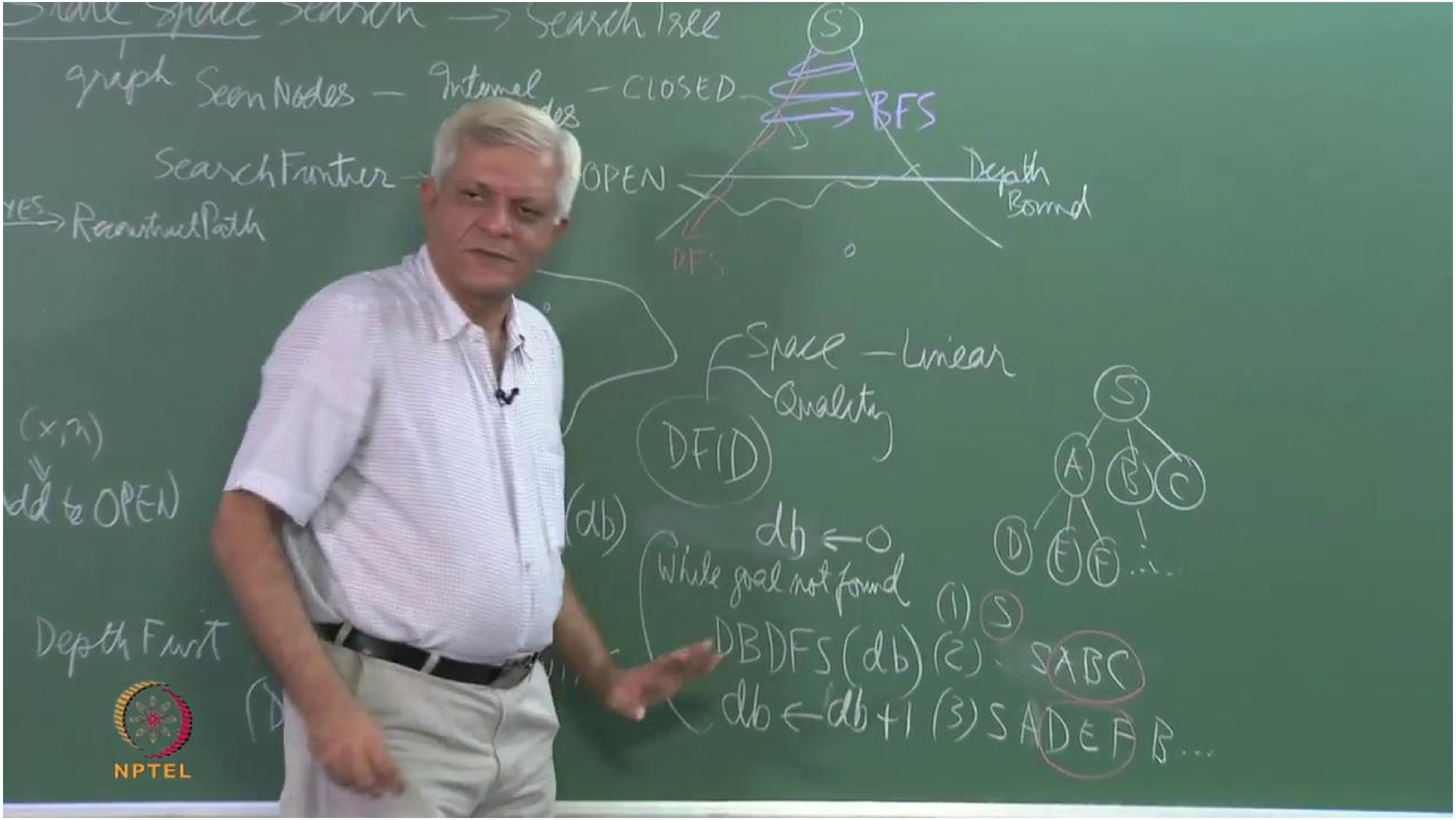
DFID

Depth Bounded DFS(db)

DBDFS(db)

ITERATIVE DEEPENING
(DFID)

```
db ← 0
While goal not found
    DBDFS(db)
    db ← db+1
```



```
db ← 0
While goal not found
    DBDFS(db)
    db ← db+1
```

(1) S

(2) SABC

(3) SADEFB...

Space – Linear

Quality

DFID

S(db)

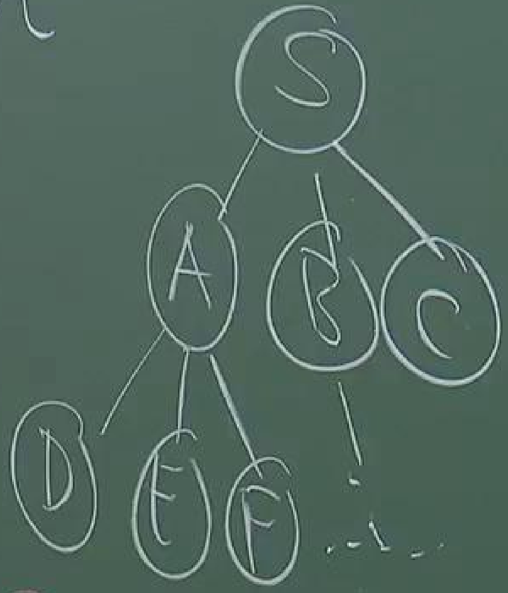
WING

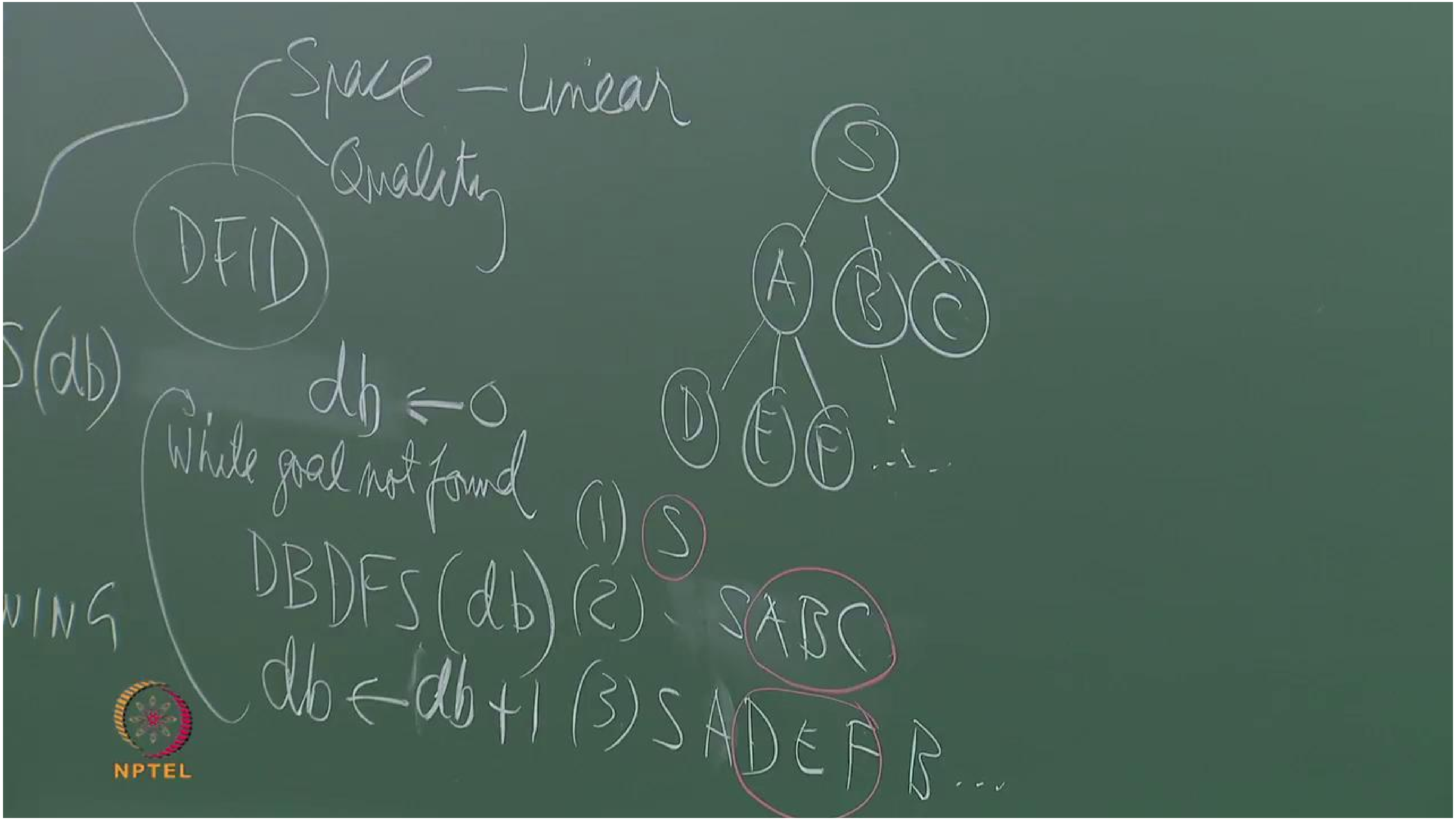
```
db ← 0
While goal not found
    DBDFS(db)
    db ← db+1
```

(1) S

(2) S A B C

(3) S A D E F B ...





DFID

- Space — Linear
- Quality

S(db)

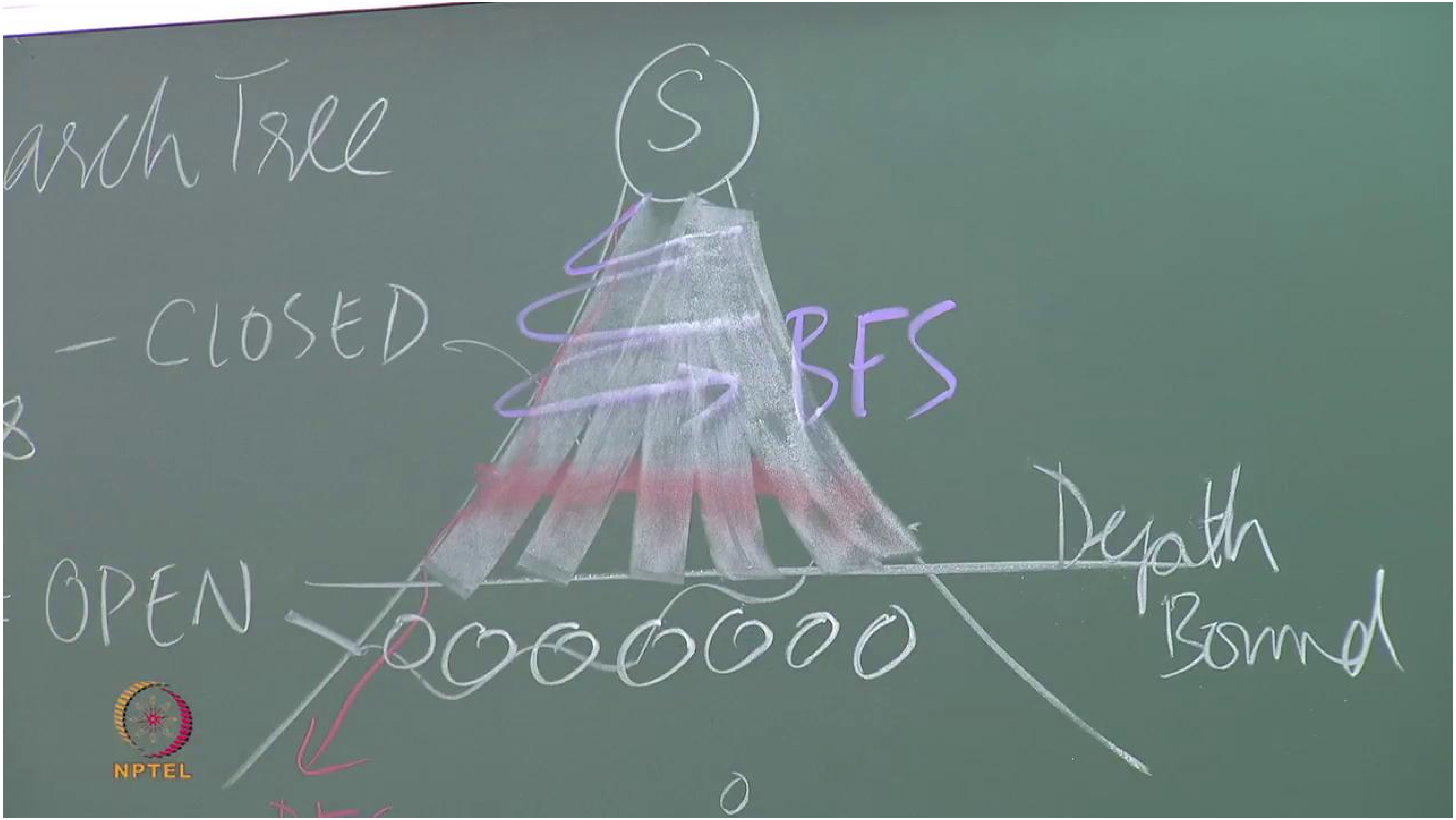
WING

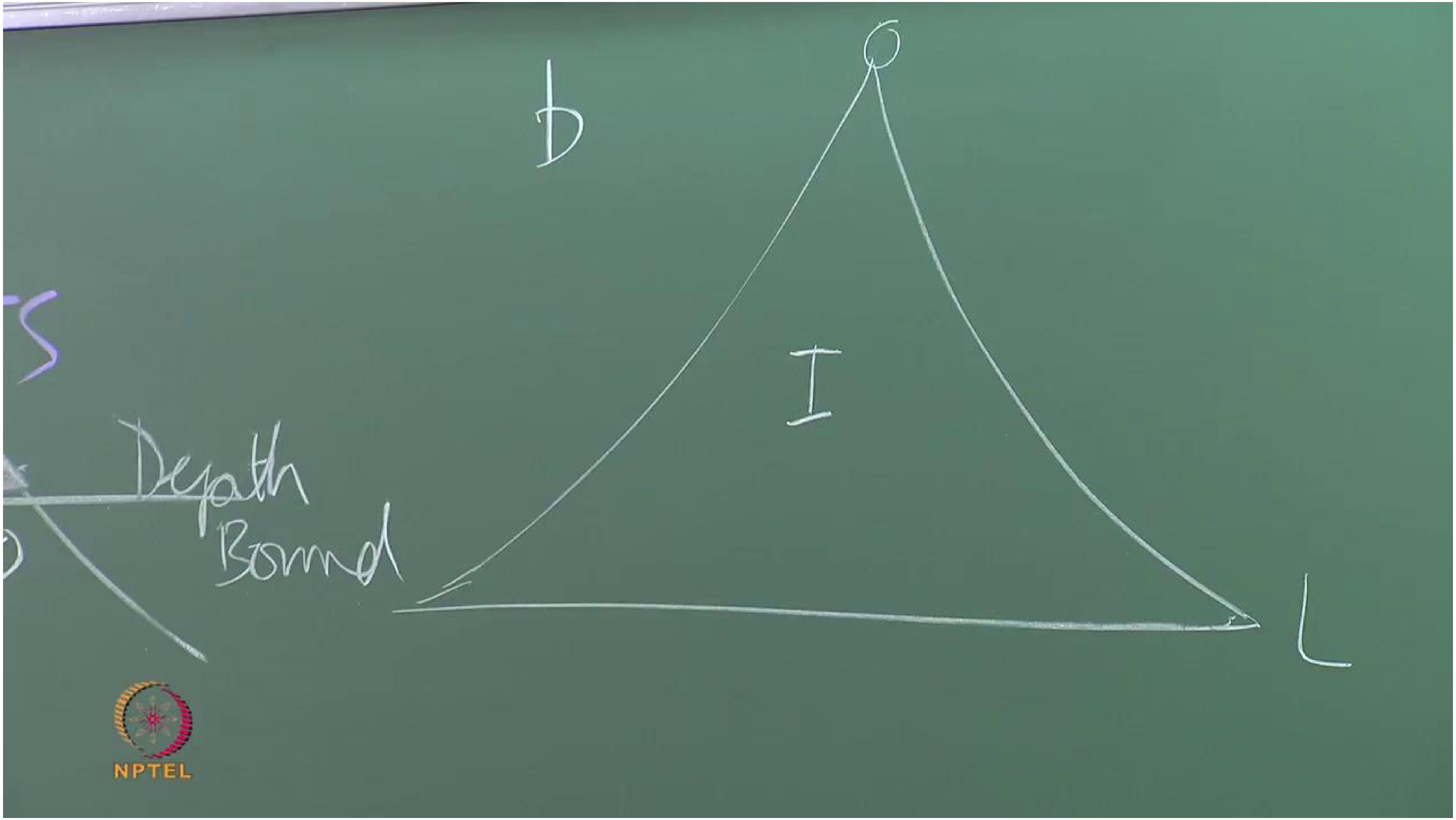
```
db ← 0
While goal not found
    DBDFS(db)
    db ← db+1
```

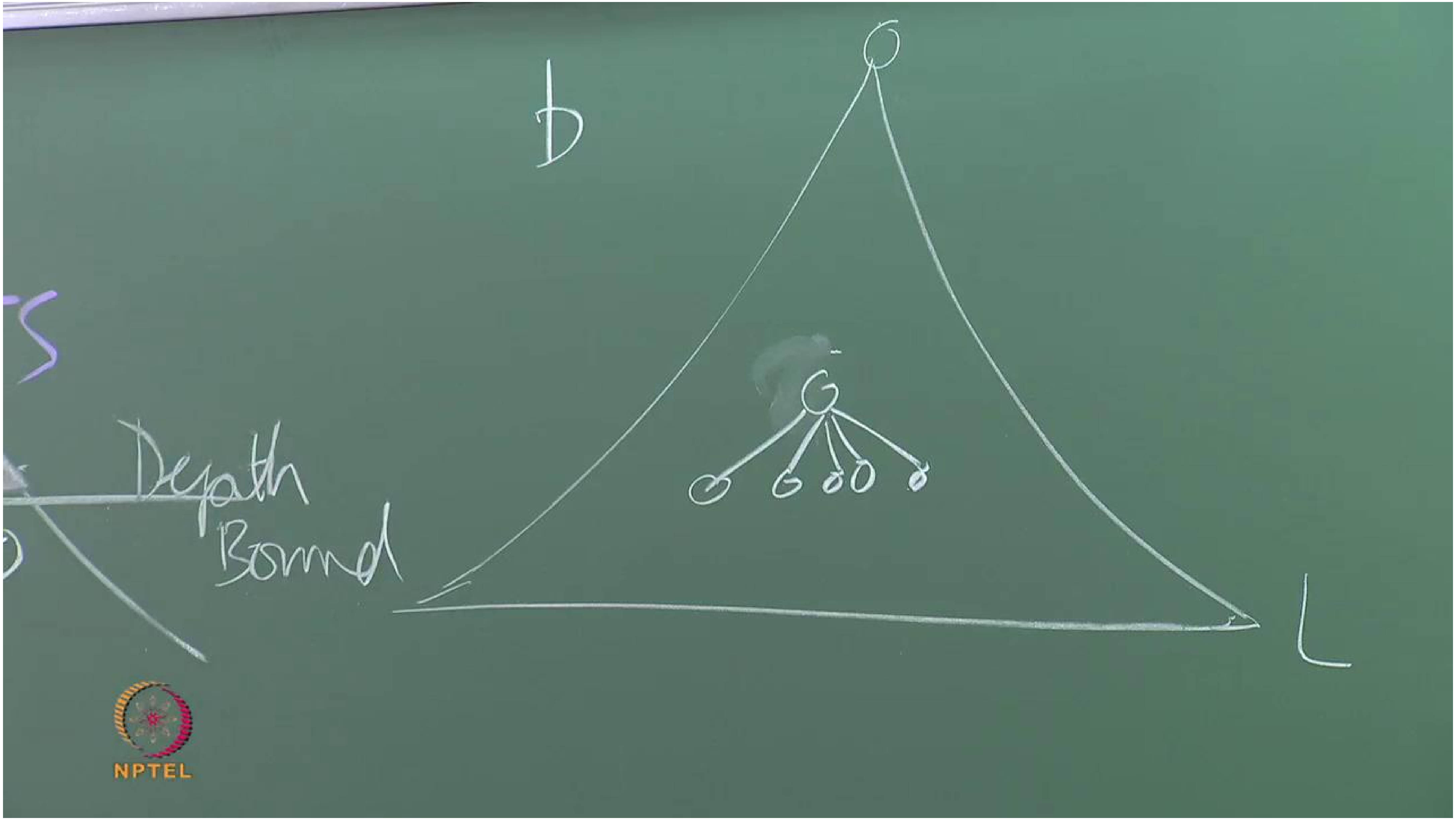
(1) S

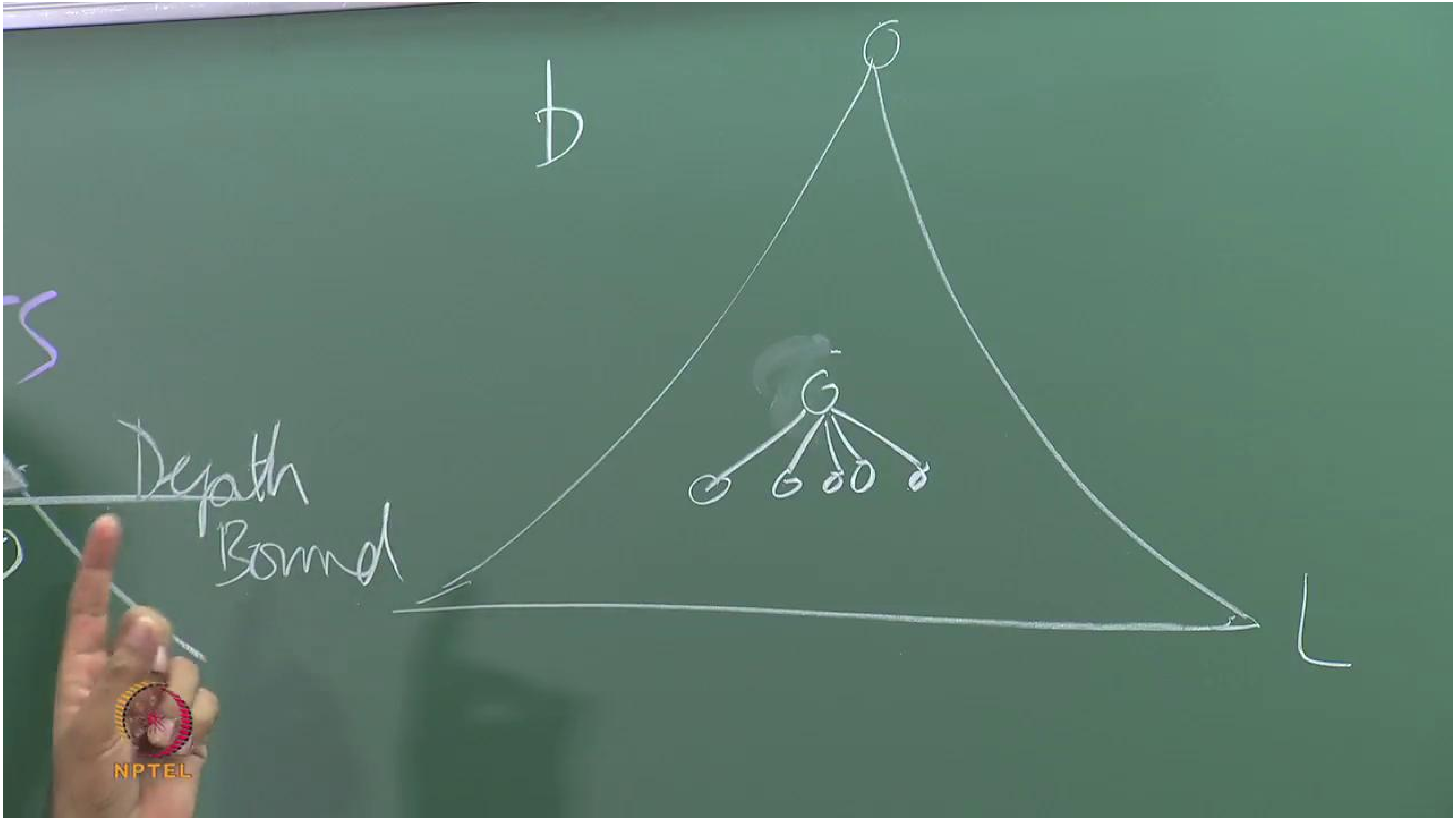
(2) S A B C

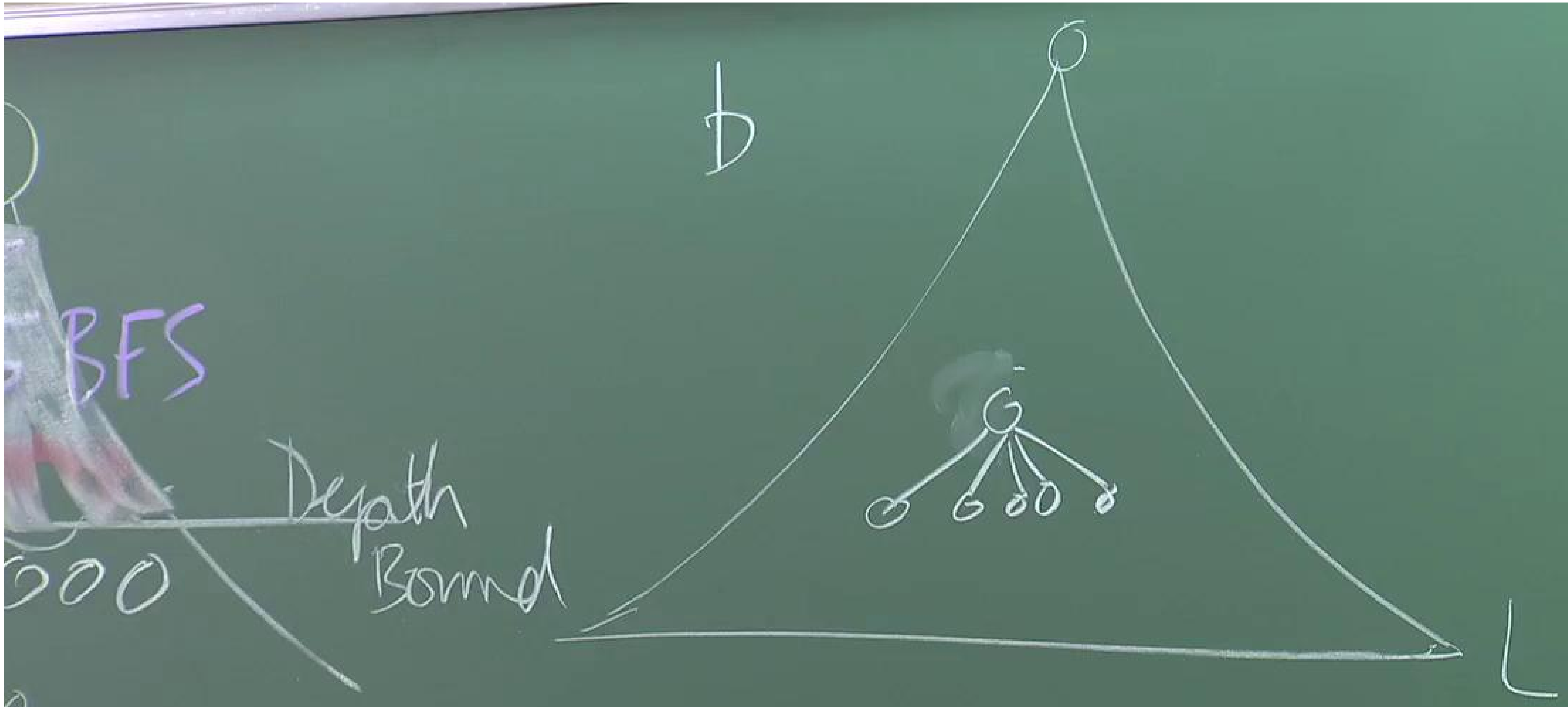
(3) S A D E F B ...


arch Tree
– CLOSED
8
OPEN
S
BFS
Depth Bound
DF


D
I
L
Depth Bound
S


b
S
G
Depth
Bound
L


b
G
Depth
Bound
L

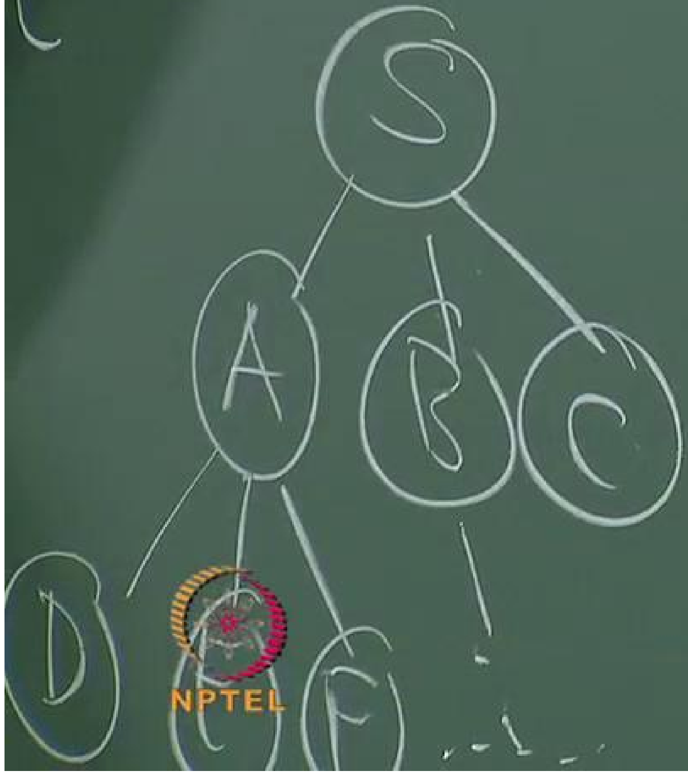


$$L = (b-1)\,I + 1$$

— Linear

$$L = (b-1)I + 1$$

$$I = \frac{L-1}{b-1}$$

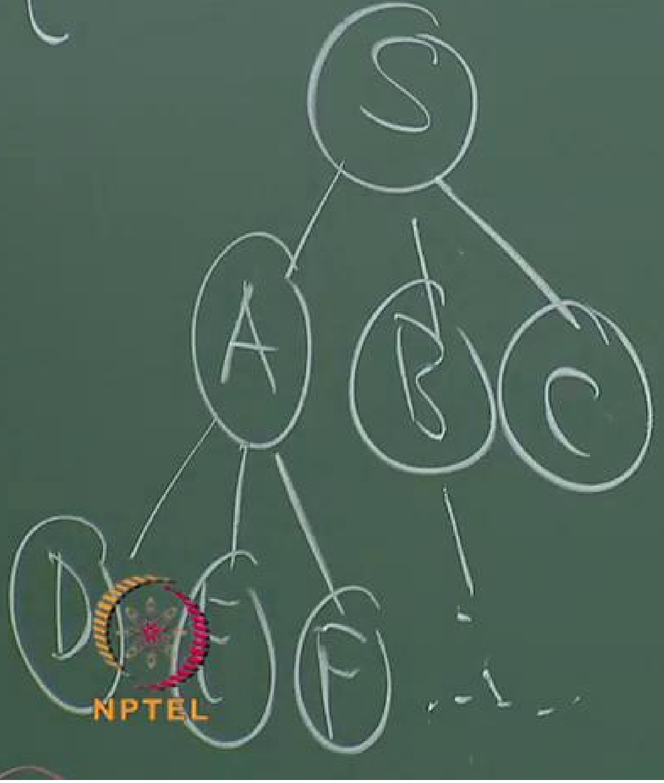


$$\frac{L+I}{L}$$

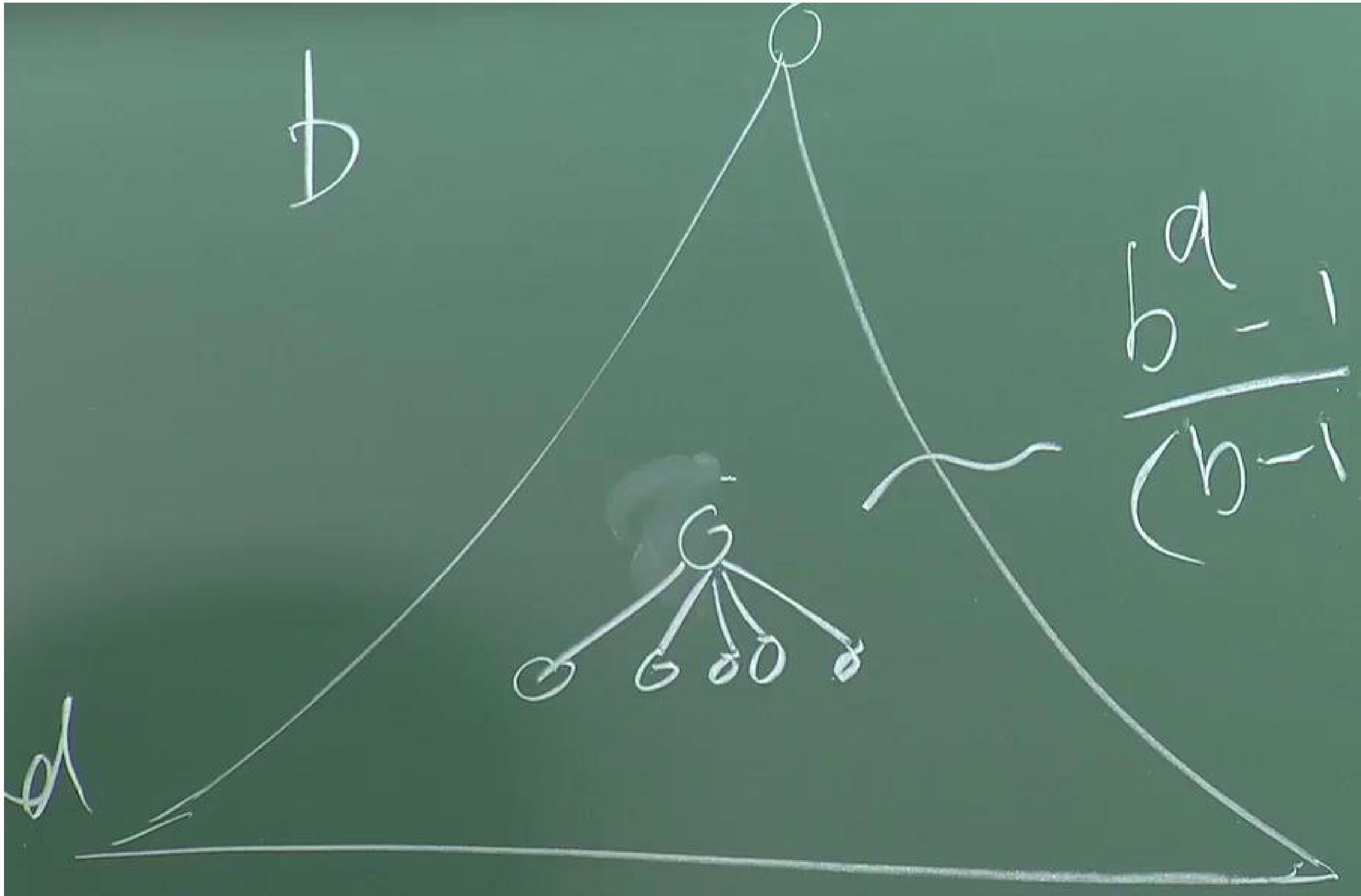
$$L = (b-1)\,I + 1$$

$$I = \frac{L-1}{b-1}$$

ar



$$\frac{L+I}{L} \approx \frac{b}{b-1}$$

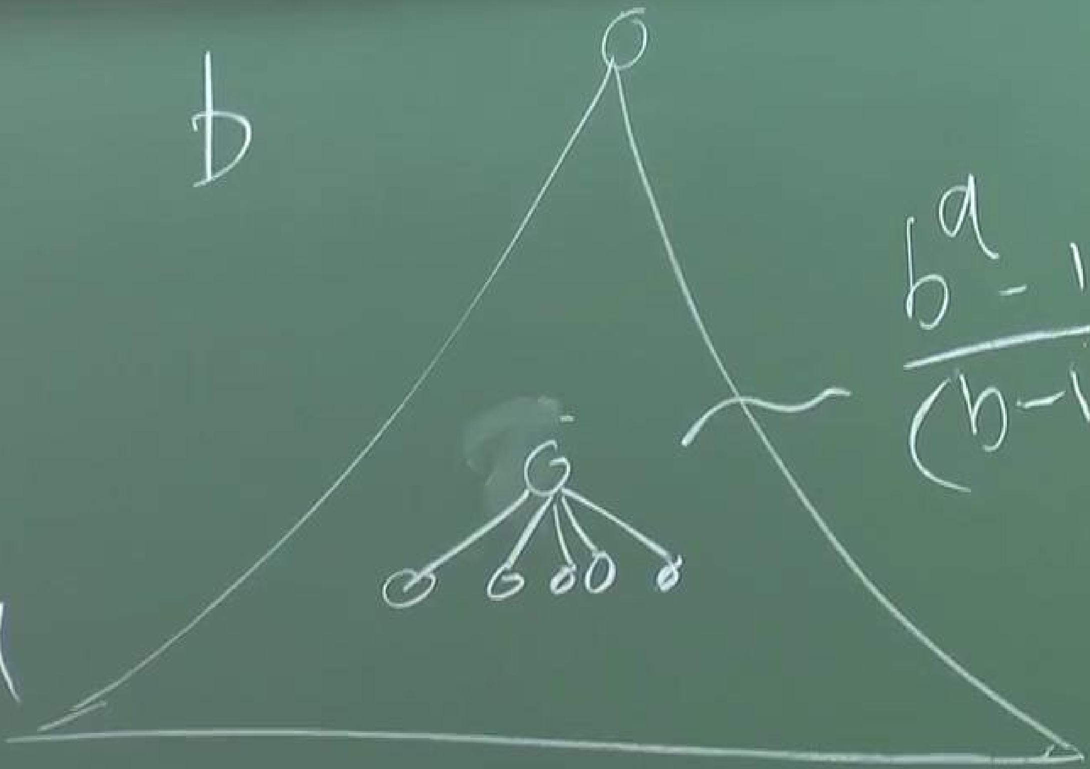


$\sim \frac{b^d - 1}{(b-1)}$

$L\ \boxed{b^d}$

$L = (b-1)\,I + 1$

b



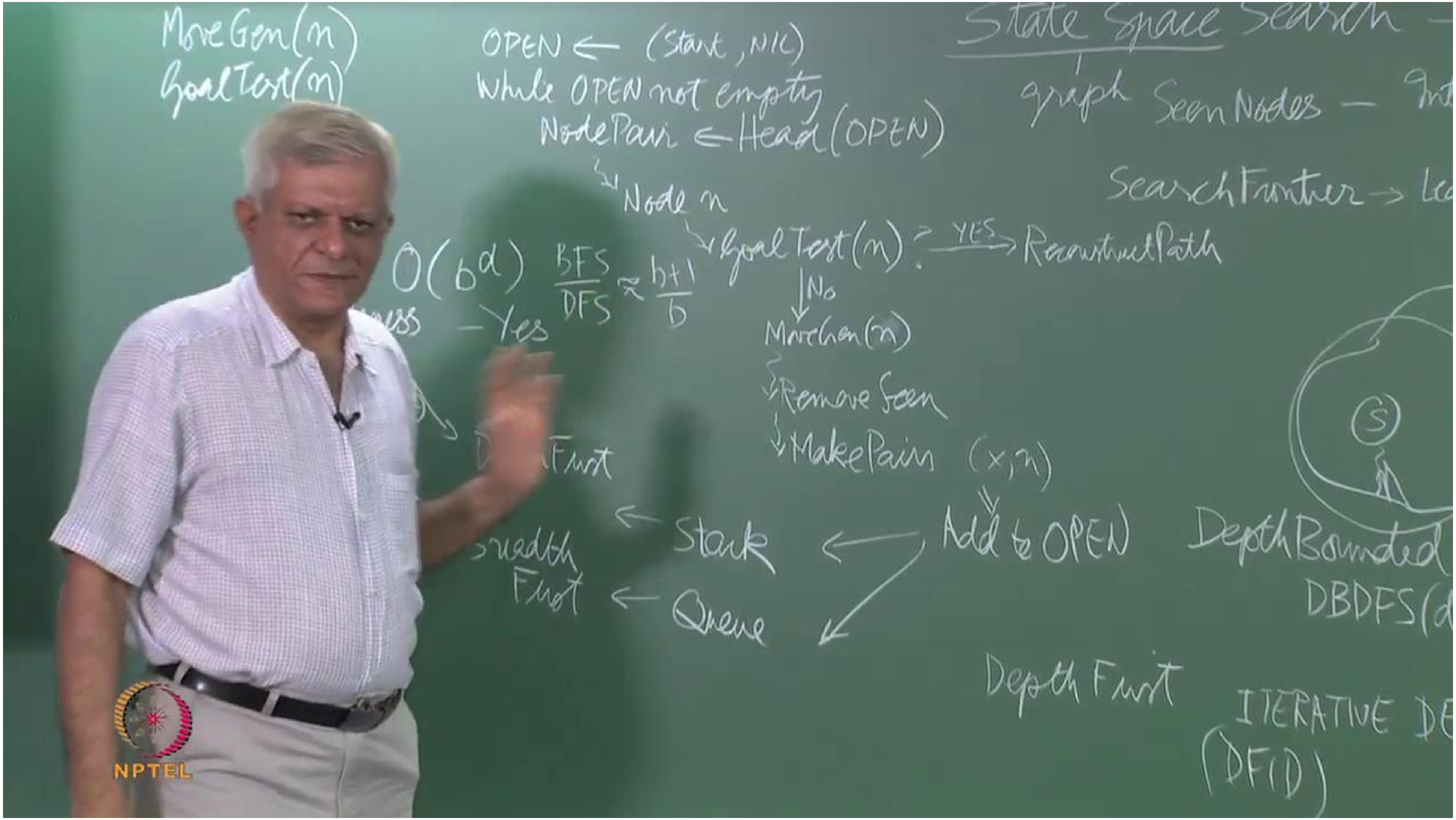
$L \approx (b-1)\,I$

$$\frac{b^d - 1}{(b-1)}$$

nd

$L\,\boxed{b^d}$

$$L = (b-1)\,I + 1$$



MoveGen(n)
GoalTest(n)

```
OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
      ↳ Node n
        ↳ GoalTest(n)? —YES→ ReconstructPath
            ↓ No
          MoveGen(n)
            ↳ Remove Seen
            ↳ MakePairs (x,n)
                ⇓
               Add to OPEN
```

Depth First ← Stack ←

Breadth First ← Queue ←

$O(b^d)$ $\frac{BFS}{DFS} \approx \frac{b+1}{b}$

…ness — Yes

State Space Search —

graph Seen Nodes — Int…

SearchFrontier → Lea…

Depth Bounded…

DBDFS(d…

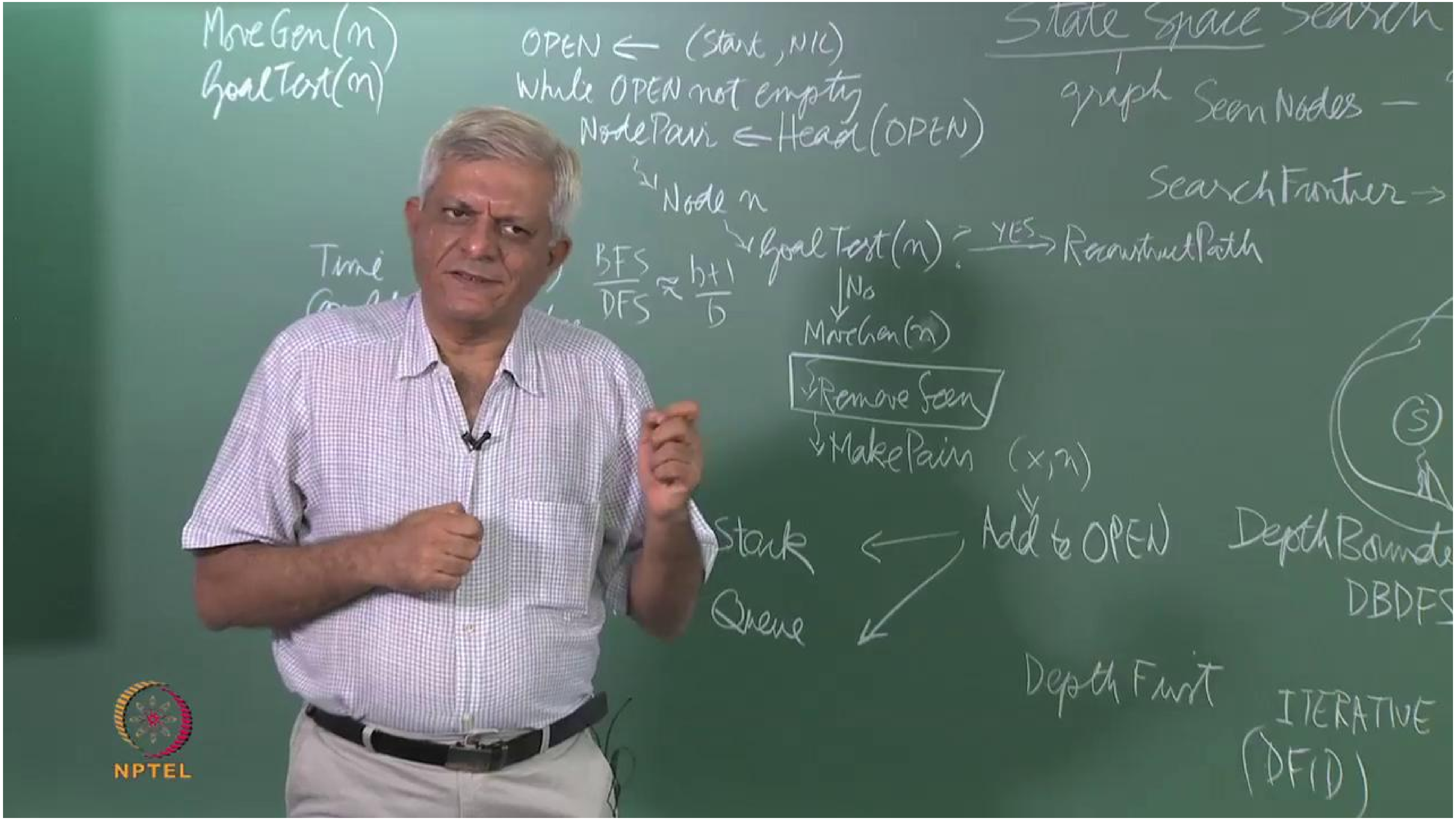
Depth First

ITERATIVE DE…
(DFID)

$b^d)$ $\frac{BFS}{DFS} \approx \frac{b+1}{b}$ ↓ goal

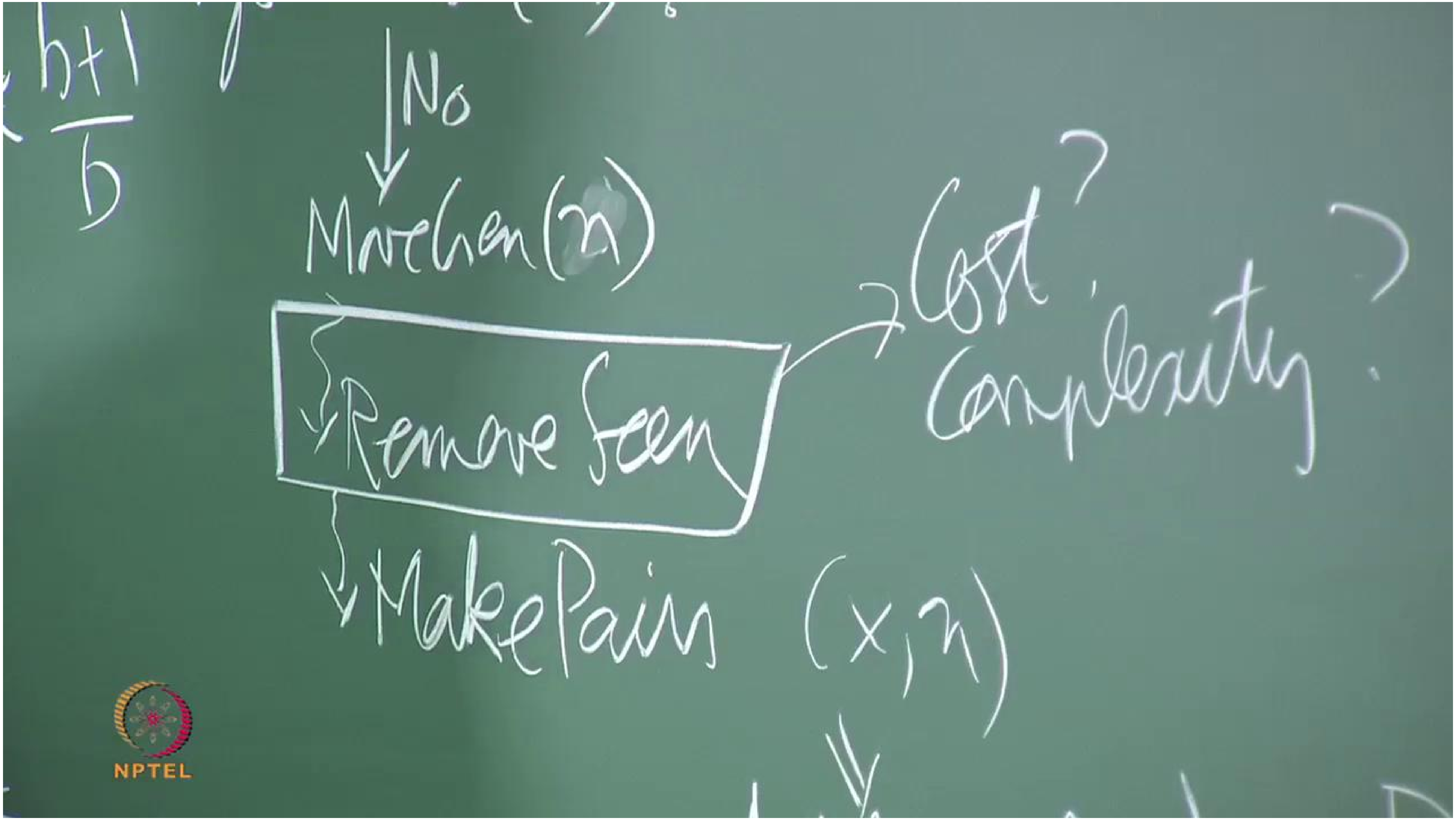
$-$ Yes

M



```
MoveGen(n)
GoalTest(n)

OPEN ← (Start, NIL)
While OPEN not empty
    NodePair ← Head(OPEN)
        Node n
            GoalTest(n) ? —YES→ ReconstructPath
                | No
                MoveGen(n)
                Remove Seen
                MakePairs (x, n)
                    Add to OPEN
```



$\frac{b+1}{b}$

No

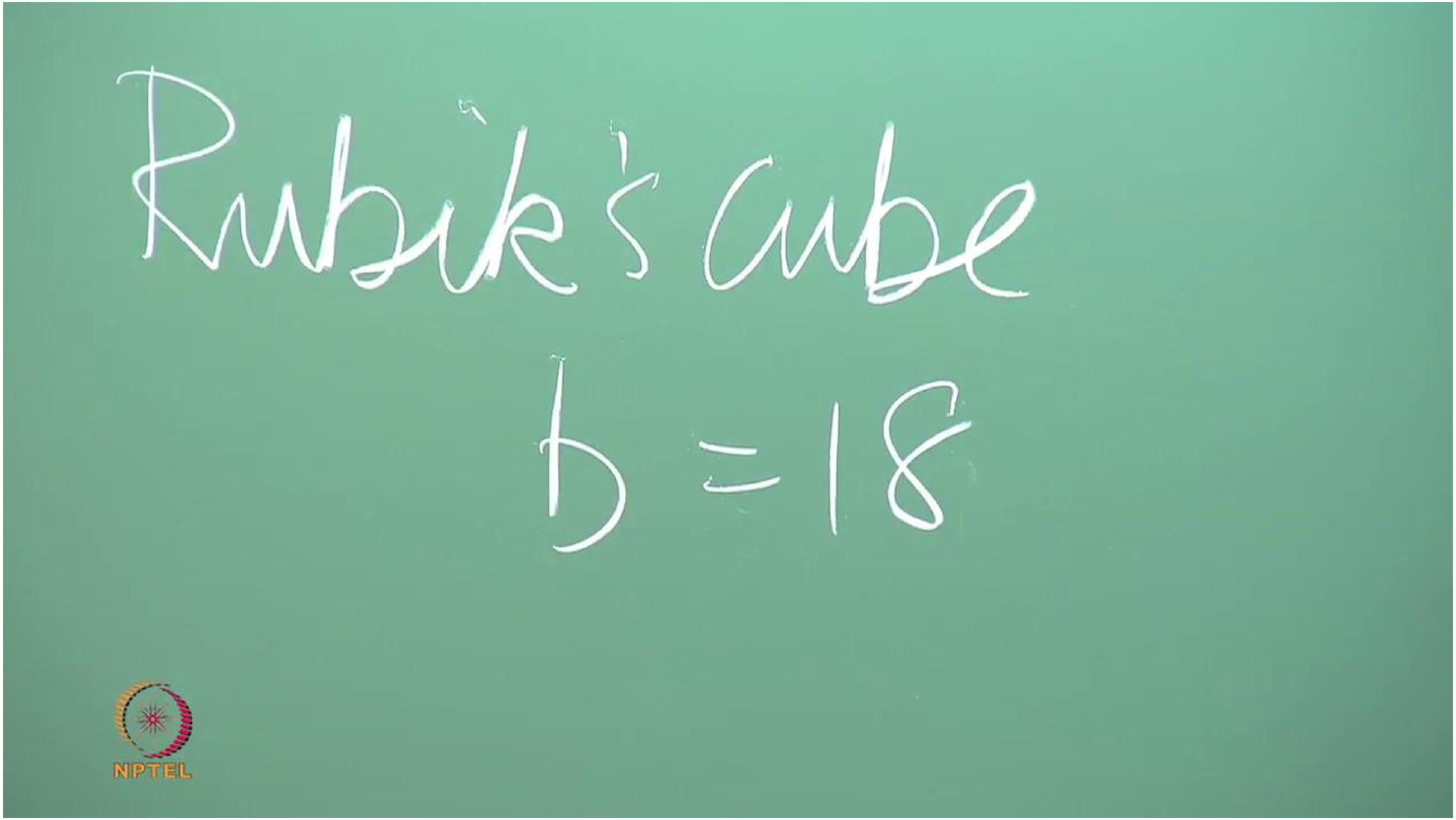
Mrechen(x)

Remove Seen

Cost?

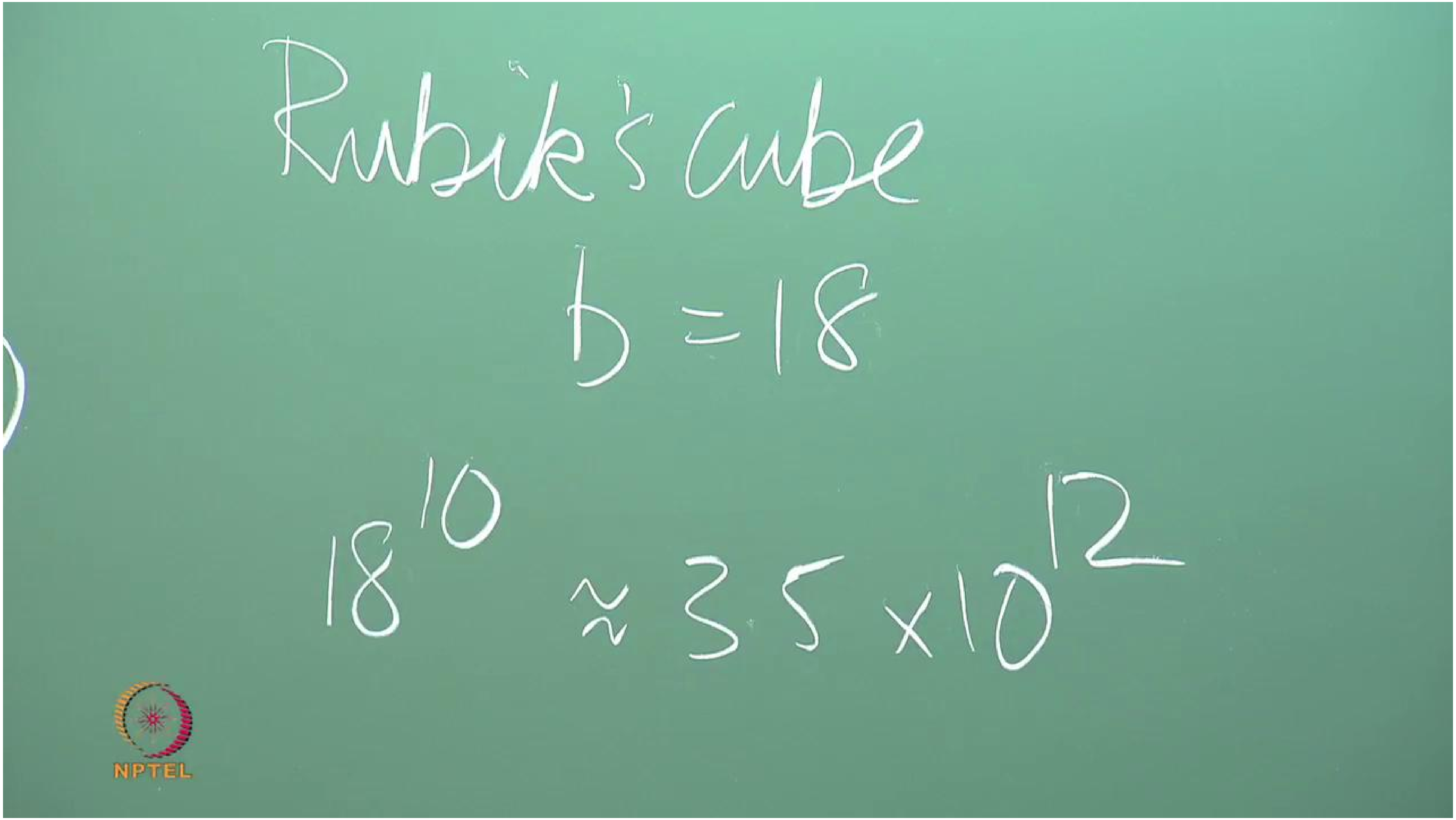
Complexity?

MakePairs (x, n)



Rubik's cube

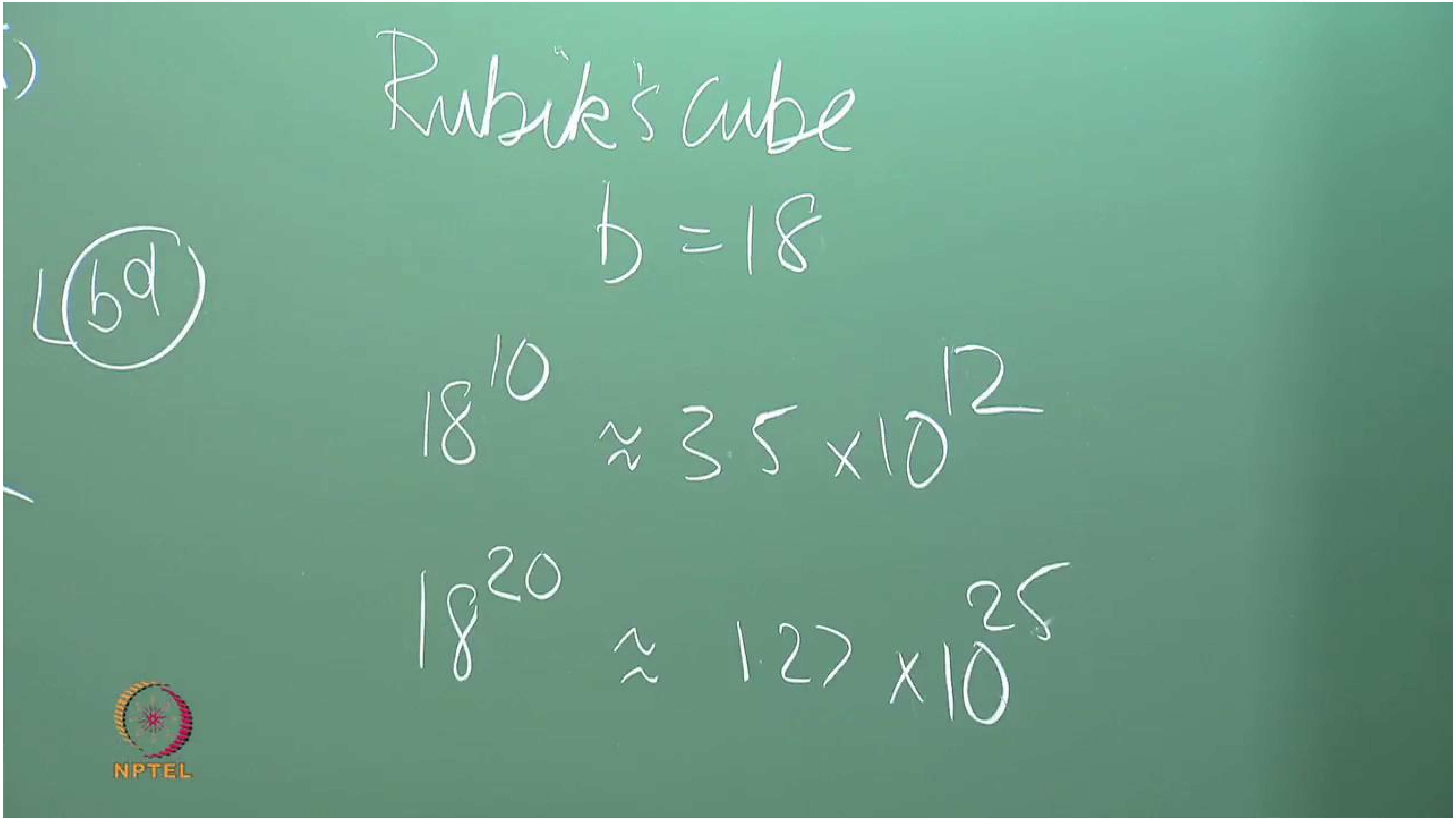
$b = 18$



Rubik's cube

$b = 18$

$18^{10} \approx 3.5 \times 10^{12}$

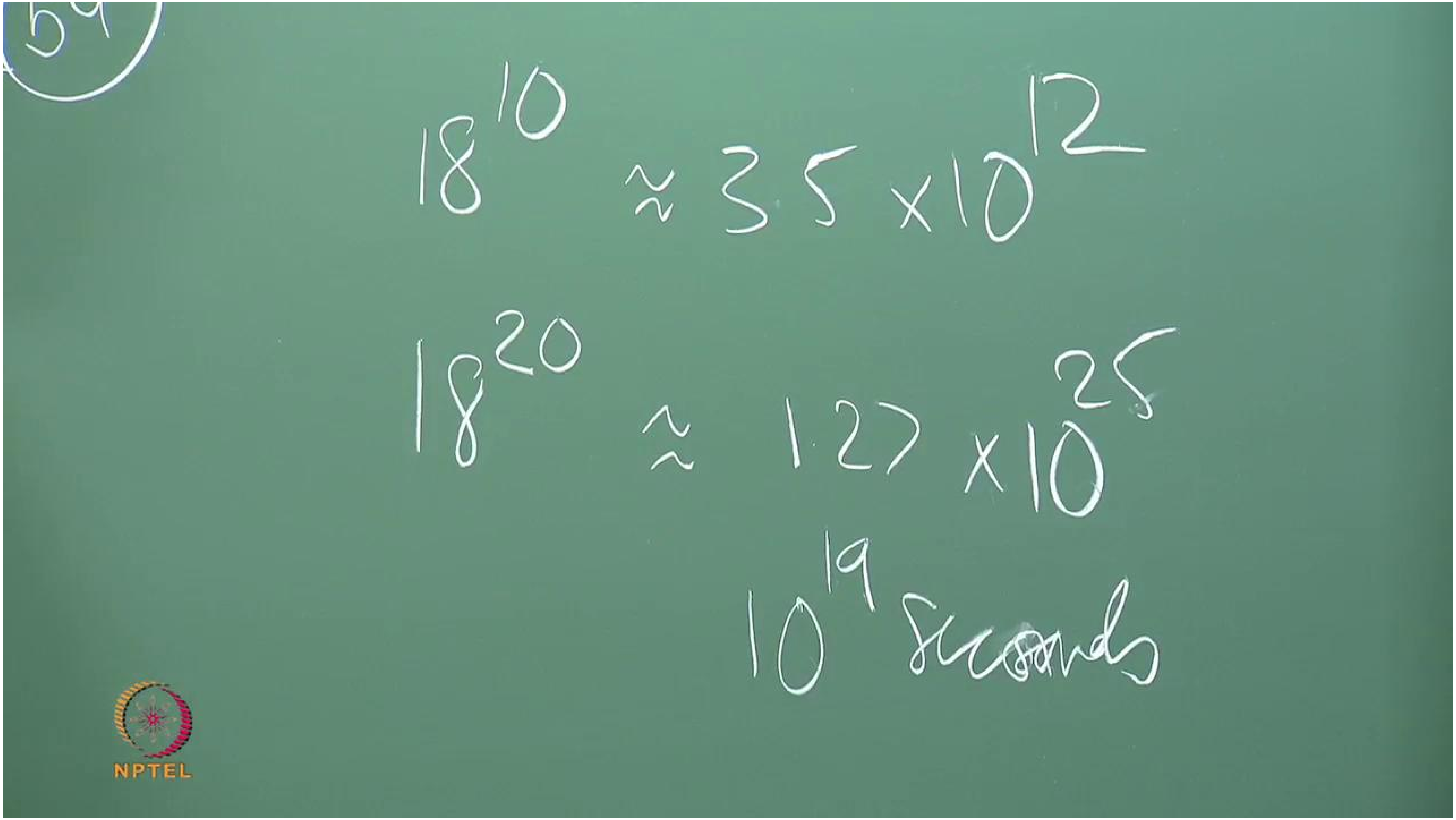


Rubik's cube

$b = 18$

$18^{10} \approx 3.5 \times 10^{12}$

$18^{20} \approx 1.27 \times 10^{25}$

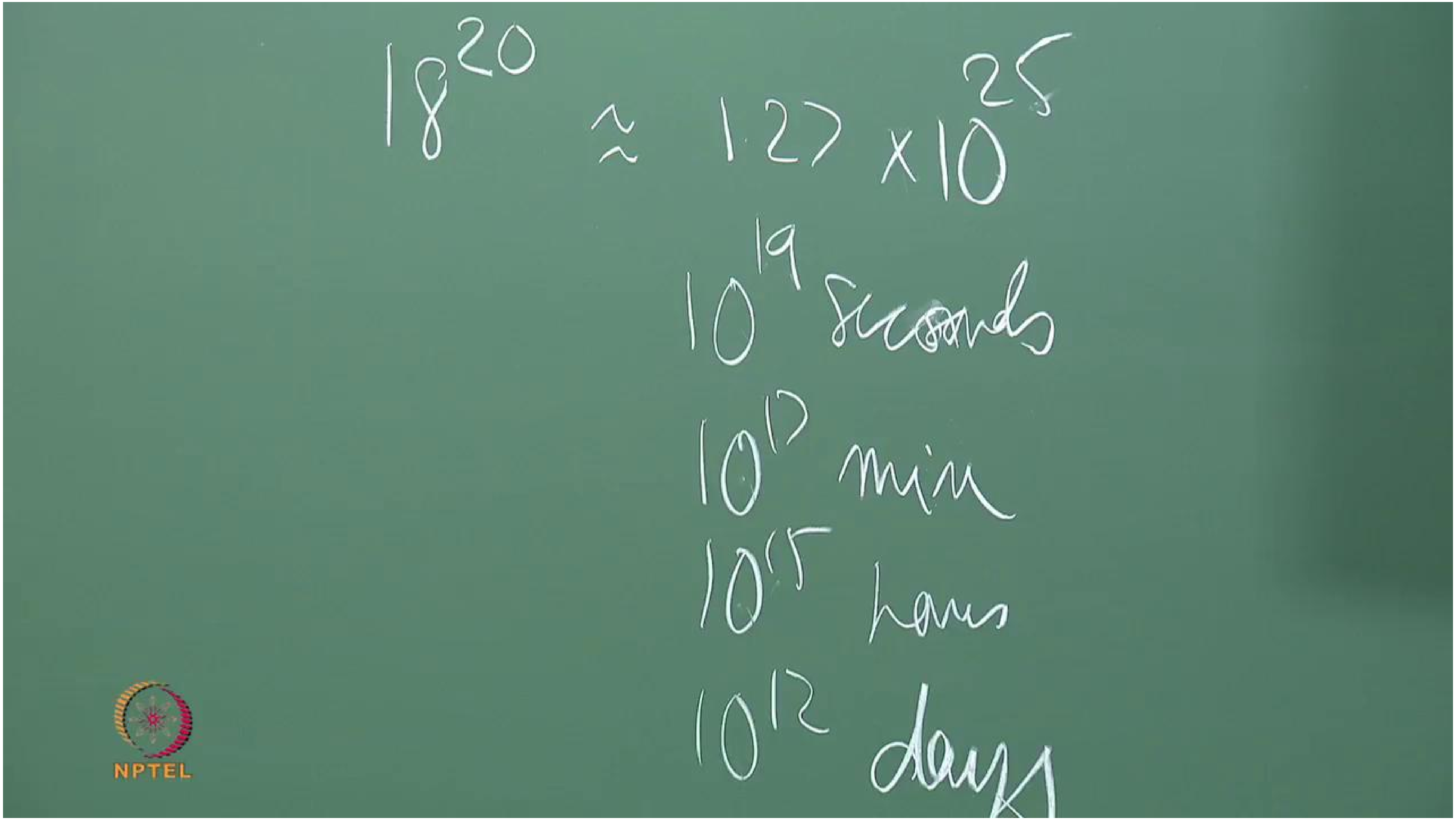


59

$$18^{10} \approx 3.5 \times 10^{12}$$

$$18^{20} \approx 1.27 \times 10^{25}$$

10^{19} seconds



$$18^{20} \approx 1.27 \times 10^{25}$$

10^{19} seconds

10^{17} min

10^{15} hours

10^{12} days

$$\approx \frac{D}{D-1}$$

$$18^{20} \approx 1.27 \times 10^{25}$$

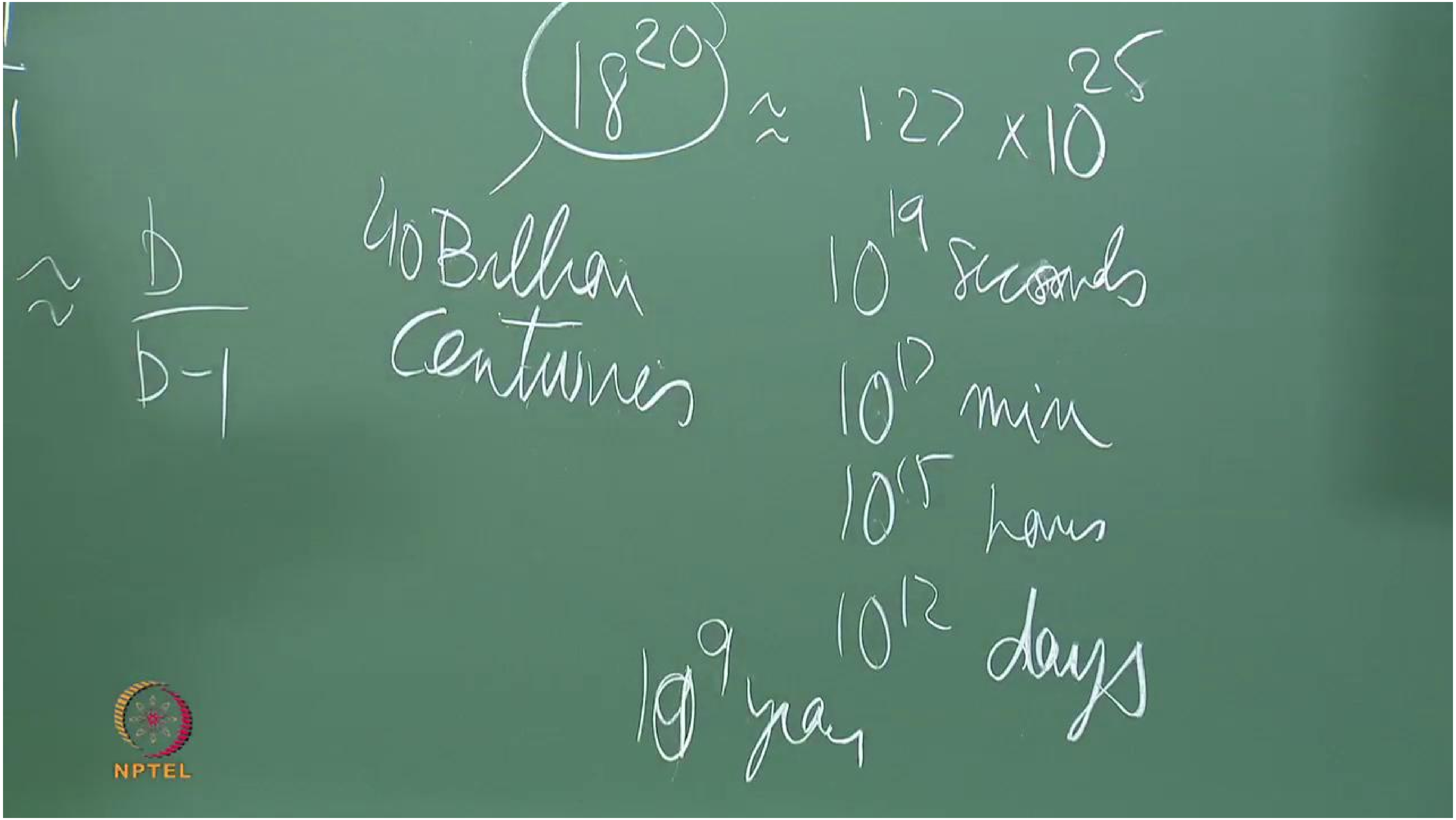
10^{19} seconds

10^{17} min

10^{15} hours

10^{12} days

10^{9} years



$$\approx \frac{D}{D-1}$$

$$\boxed{18^{20}} \approx 1.27 \times 10^{25}$$

40 Billion Centuries

10^{19} seconds

10^{17} min

10^{15} hours

10^{12} days

10^{9} years