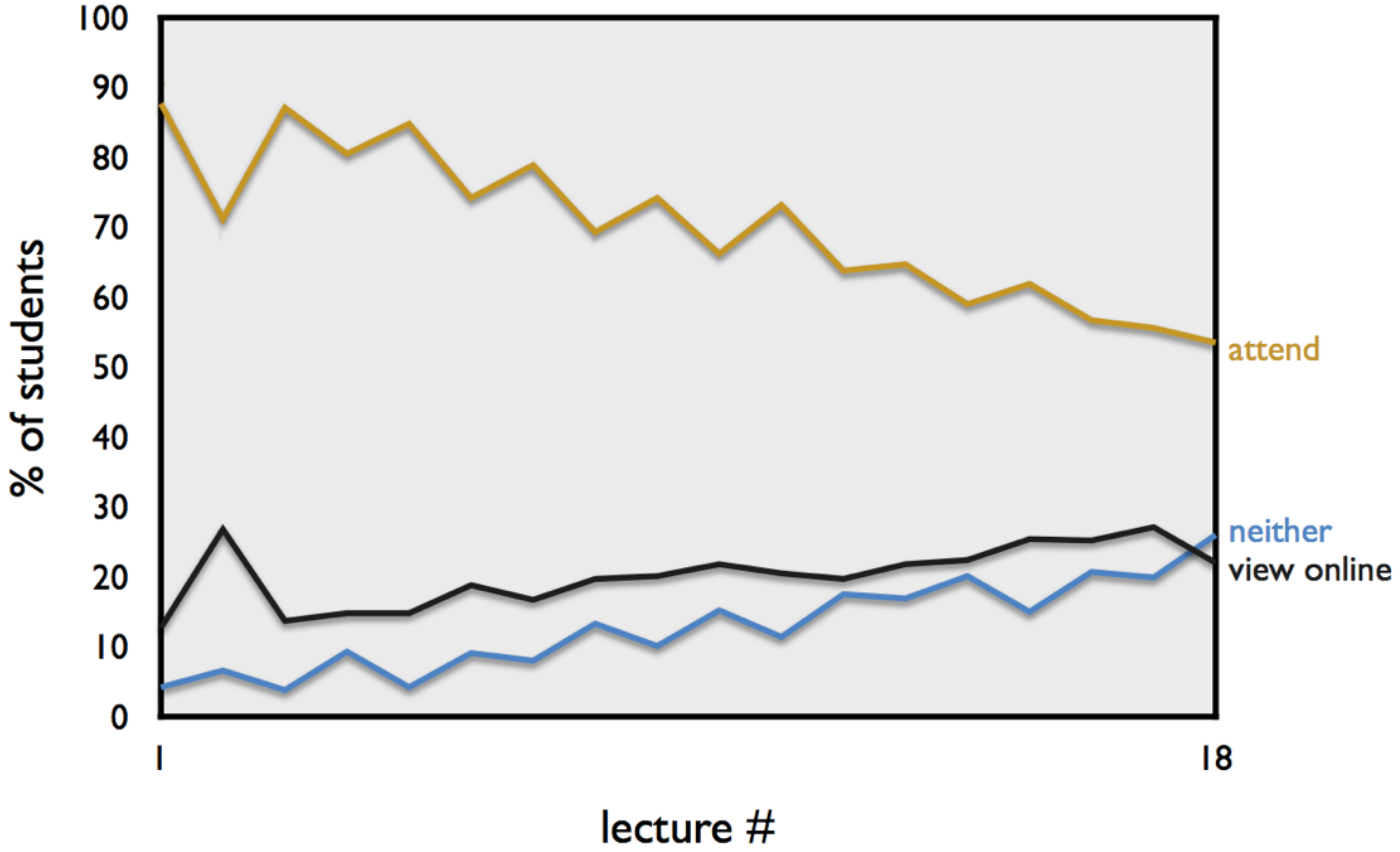
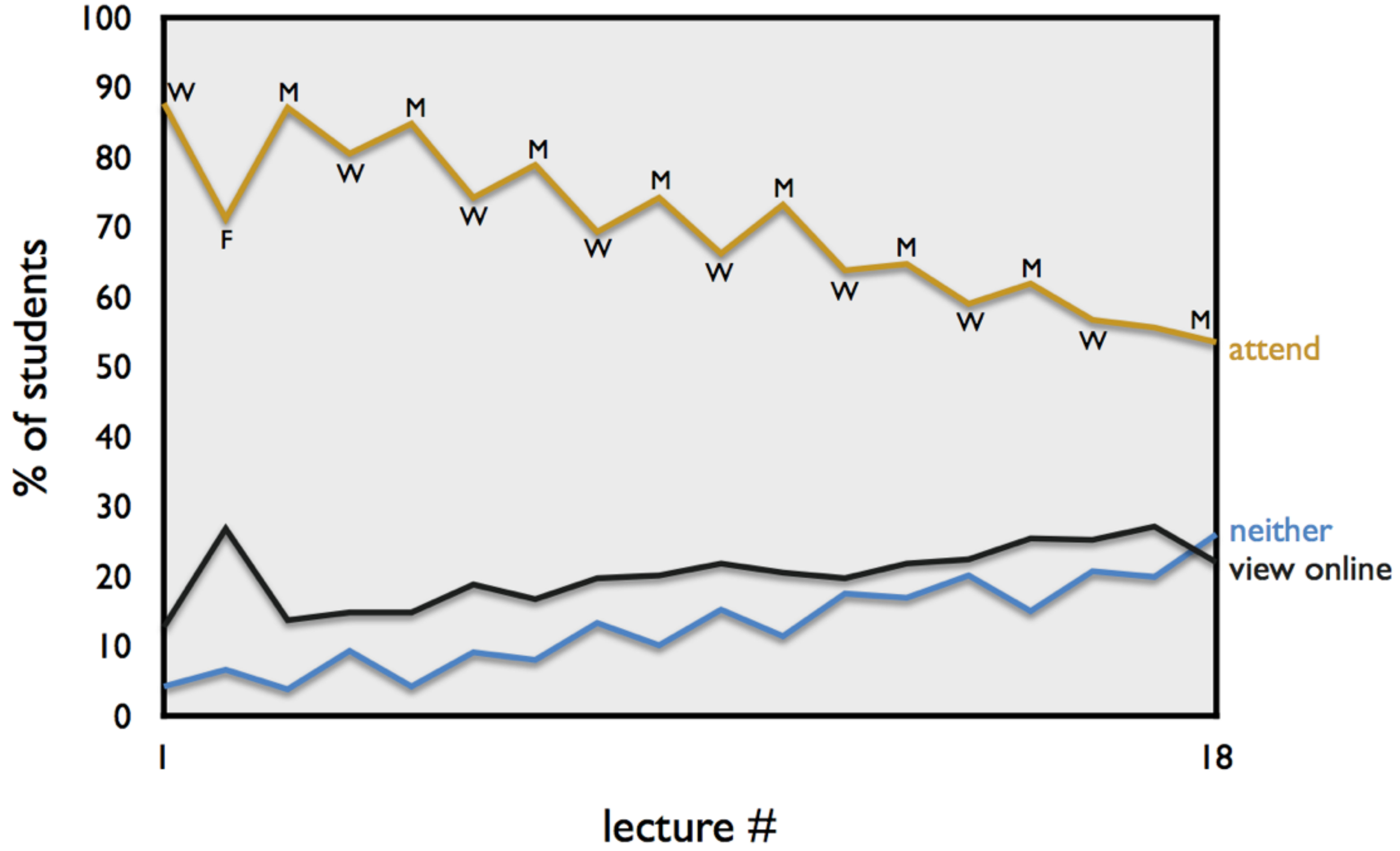
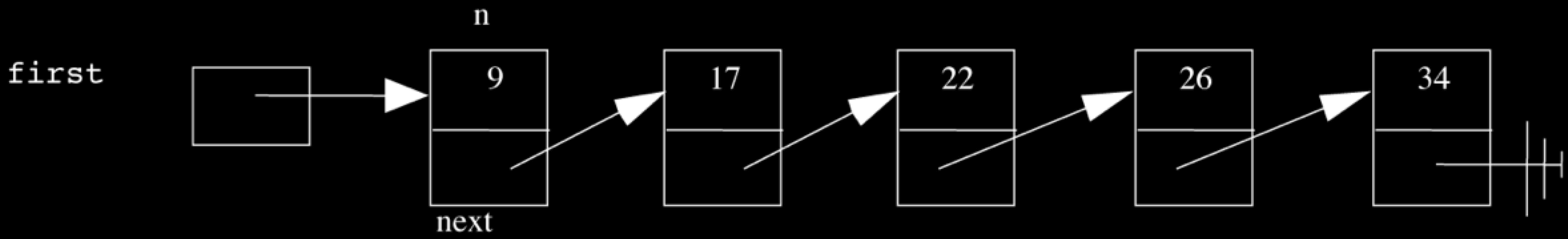
week 6


% of students
100
90
80
70
60
50
40
30
20
10
0
1
18
lecture #
attend
neither
view online



100
90
80
70
60
50
40
30
20
10
0
% of students
1
18
lecture #
W
F
M
W
M
W
M
W
M
W
M
W
M
W
M
W
M
attend
neither
view online

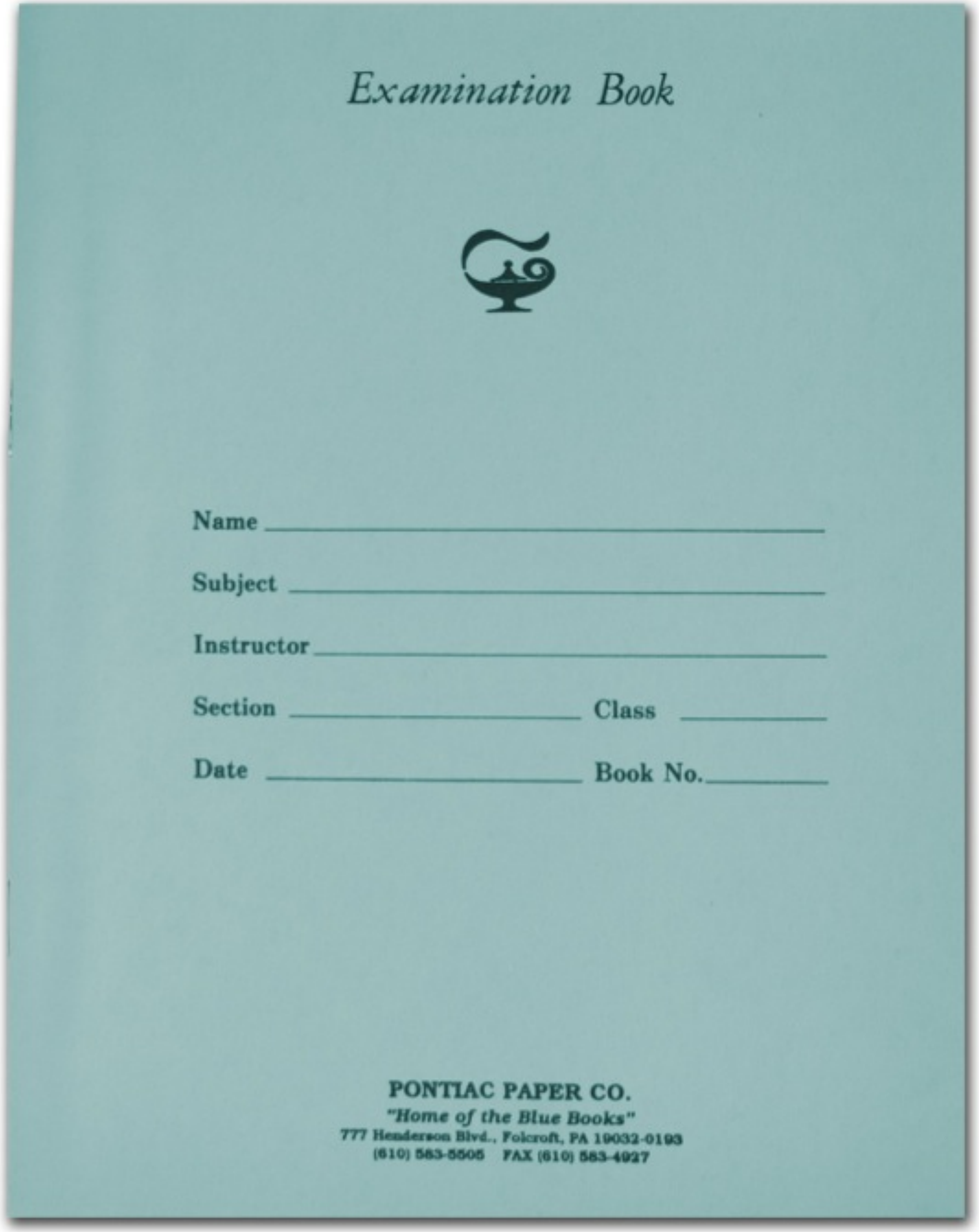


http://cs.calvin.edu/books/c++/ds/1e/

$O(n)$

$\Omega(1)$

$O(1)$



Examination Book

Name ______________________________

Subject ____________________________

Instructor __________________________

Section ______________ Class ________

Date ________________ Book No. ______

PONTIAC PAPER CO.

"Home of the Blue Books"

777 Henderson Blvd., Folcroft, PA 19032-0193

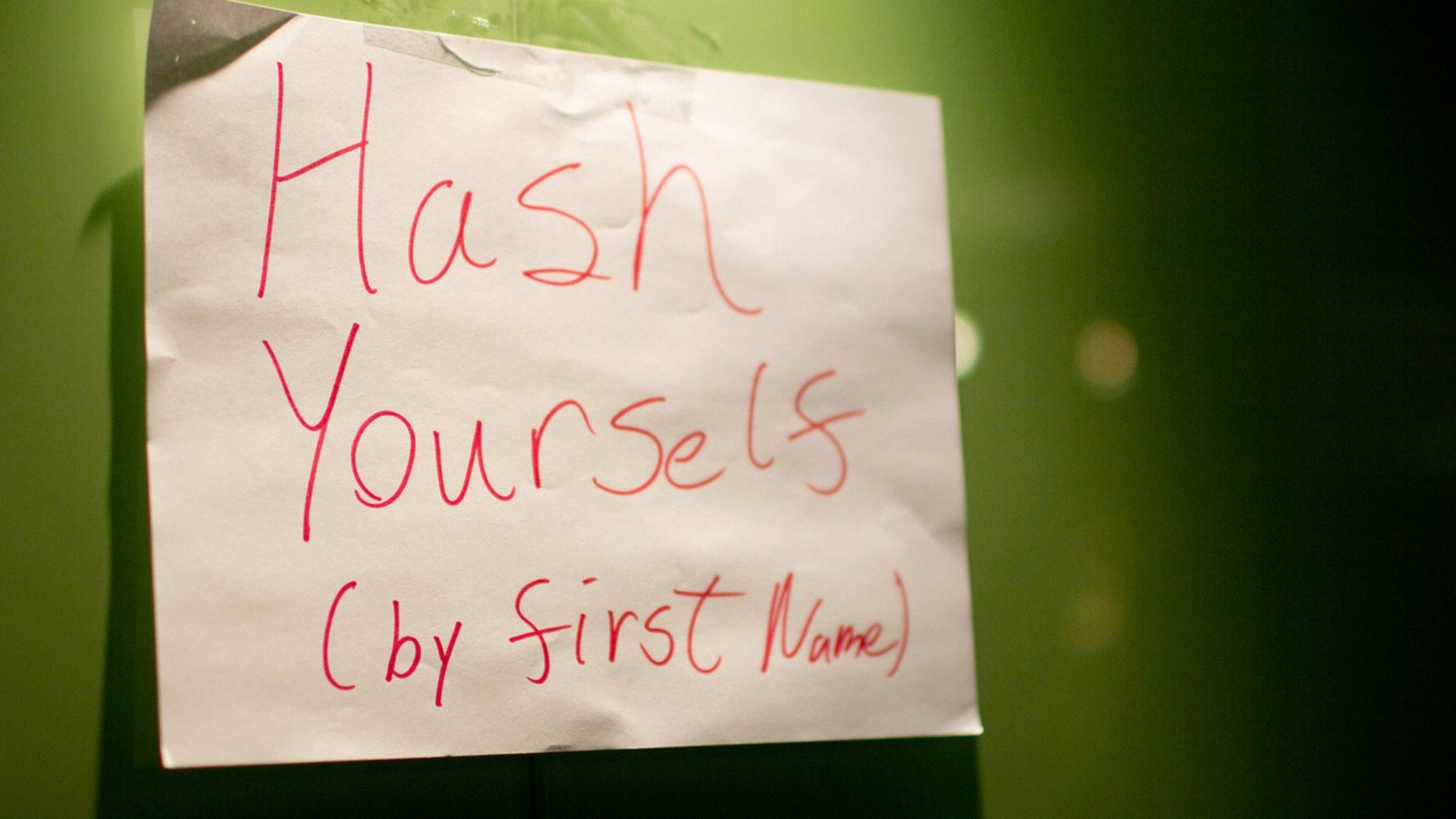
(610) 583-5505 FAX (610) 583-4927

Circle your teaching fellow's name.

Aidi Zhang	Emily Houlihan	Kevin Mu	Rob Bowden
Alex Pong	Eric Ouyang	Lily Tsai	Robbie Gibson
Allison Buchholtz-Au	Frederick Widjaja	Luciano Arango	Saheela Ibraheem
Ankit Gupta	Gabriel Guimaraes	Luis Perez	Sam Green
Armaghan Behlum	Gal Koplewitz	Lukas Missik	Stephen Turban
Arvind Narayanan	George Lok	Marcus Powers	Theo Levine
Belinda Zeng	Hannah Blumberg	Mehdi Aourir	Tiffany Wu
Camille Rekhson	Ian Nightingale	Michael Patterson	Tim McLaughlin
Chris Lim	Jackson Steinkamp	Michelle Danoff	Tomas Reimers
Cynthia Meng	Jason Hirschhorn	Nicholas Larus-Stone	Tony Ho
Dan Bradley	Jonathan Miller	Nick Joseph	Vipul Shekhawat
Daven Farnham	Jordan Canedy	Nick Mahlangu	Wellie Chao
David Kaufman	Joshua Meier	Rei Otake	Wesley Chen
Doug Lloyd	Keenan Monks	Rhed Shi	Willy Xiao
			Winnie Wu

ITT
1787

PROPERTY
SCS
STAFF

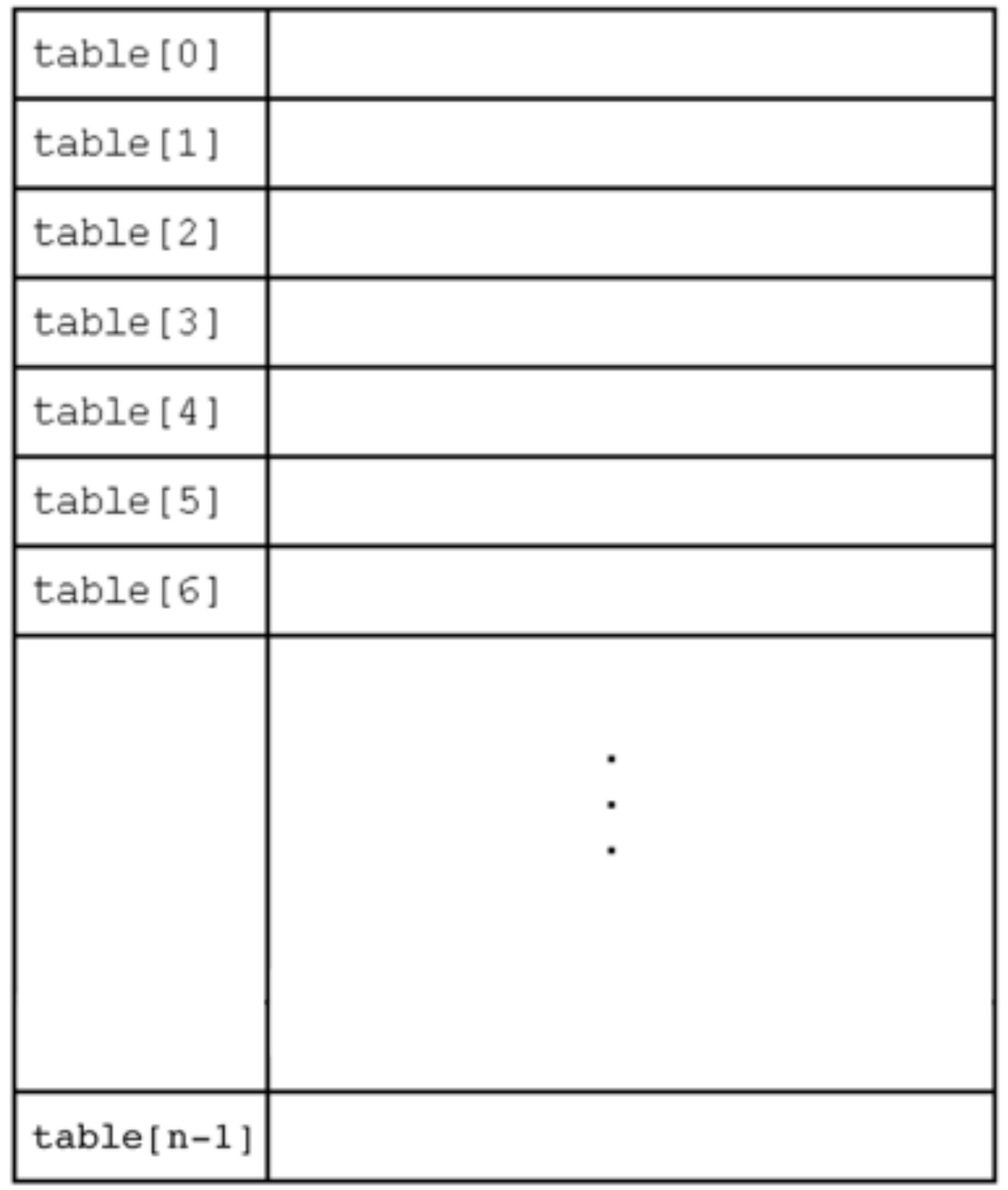

Hash Yourselfs
(by first Name)

hash table

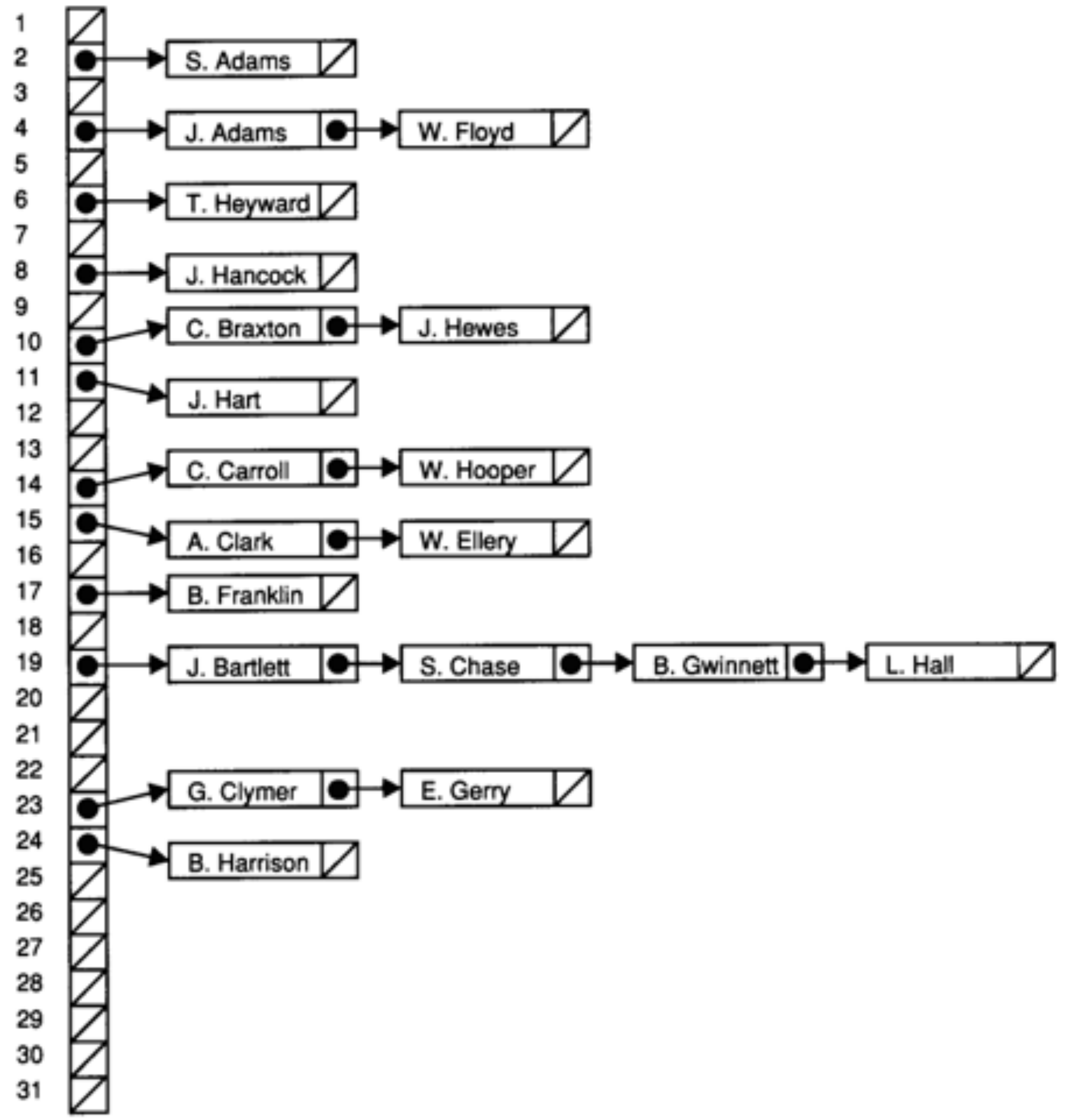
table[0]	
table[1]	
table[2]	
table[3]	
table[4]	
table[5]	
table[6]	
	⋮
table[24]	
table[25]	

```
char* table[CAPACITY];
```

linear probing



separate chaining



```
typedef struct node
{
    char* word;
    struct node* next;
}
node;
```

```c
typedef struct node
{
    char* word;
    struct node* next;
}
node;

node* table[CAPACITY];
```

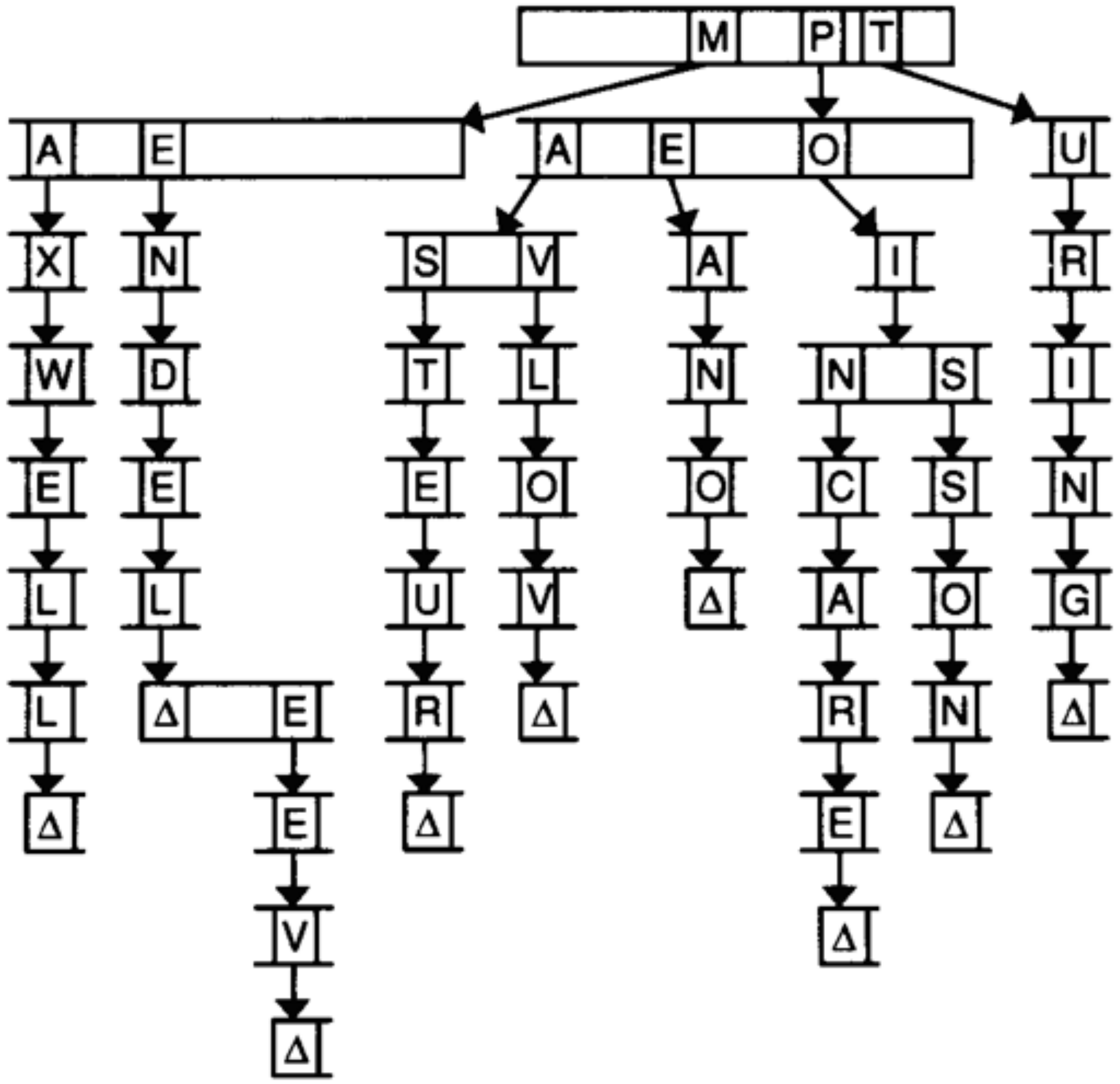


Figure from Lewis and Denenberg's Data Structures & Their Algorithms.

```
typedef struct node
{
    bool word;
    struct node* children[27];
}
node;

node* trie;
```

THANKS FOR YOUR HELP

stack

push, pop

stack

last in first out
(LIFO)

```
typedef struct node
{
    int number;
    struct node* next;
}
node;
```

```
typedef struct node
{
    int number;
    struct node* next;
}
node;

node* stack;
```

http://www.blogcdn.com/www.engadget.com/media/2008/05/iphone_line_1-1.jpg

queue

enqueue, dequeue

queue

first in first out
(FIFO)

tree

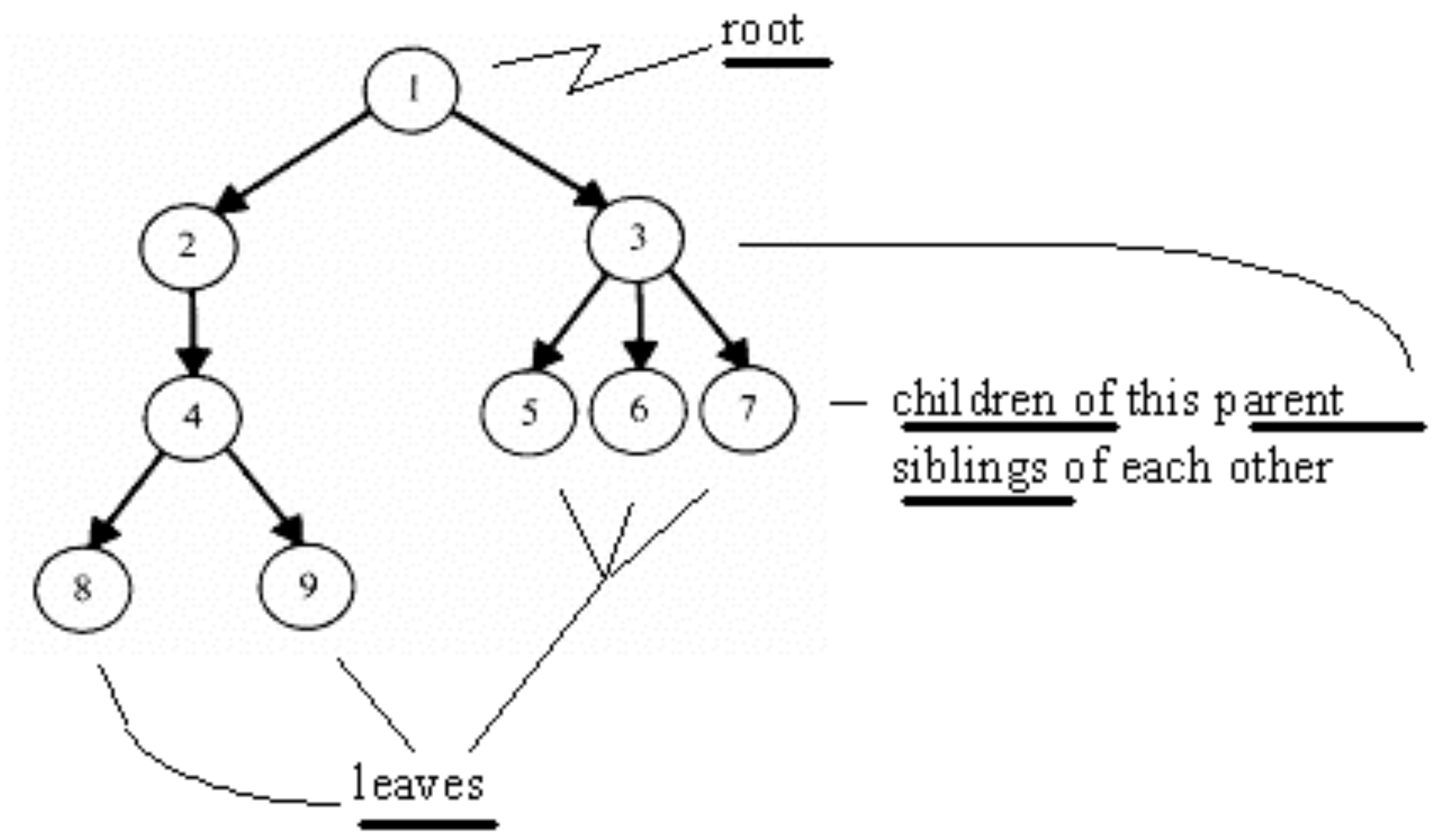


Figure by Larry Nyhoff.

binary search tree

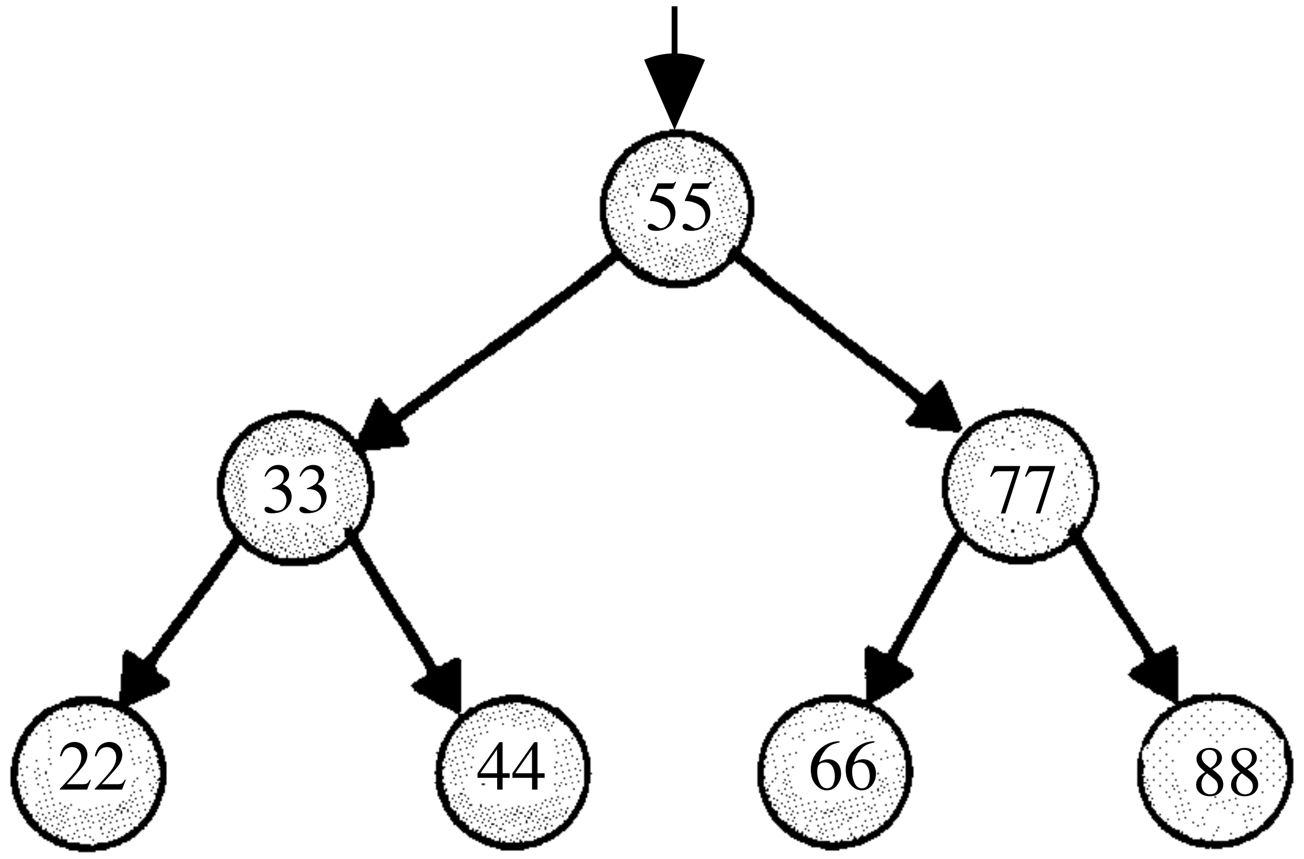
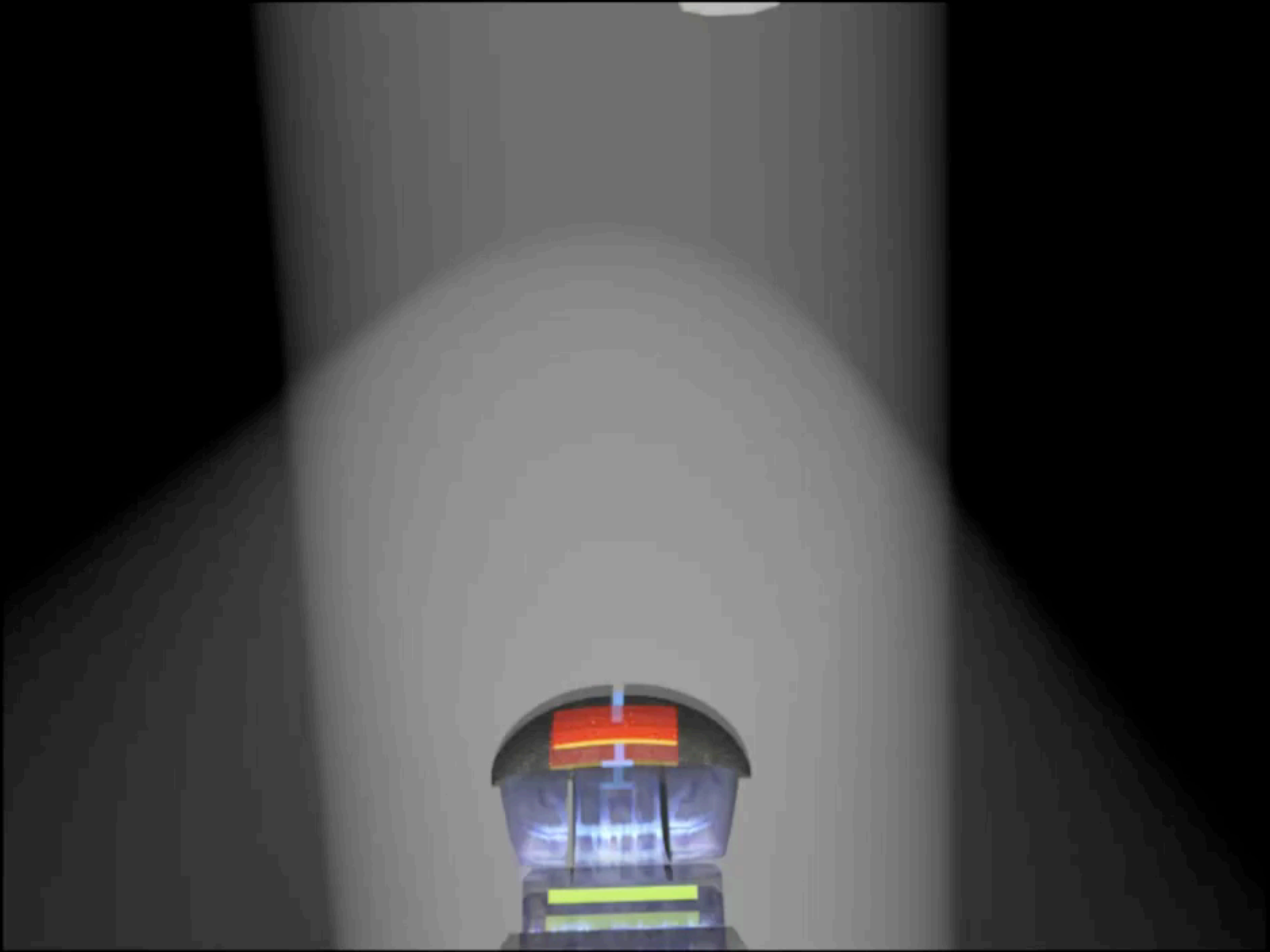


Figure from http://cs.calvin.edu/books/c++/ds/1e/.

```
typedef struct node
{
    int n;
    struct node* left;
    struct node* right;
}
node;
```

```
bool search(int n, node* tree)
{
    if (tree == NULL)
    {
        return false;
    }
    else if (n < tree->n)
    {
        return search(n, tree->left);
    }
    else if (n > tree->n)
    {
        return search(n, tree->right);
    }
    else
    {
        return true;
    }
}
```



to be continued...