

CS 189 today

A few more problems... ... and one more algorithm!

Open-ended project option...

Jotto game finale?!

First ones have
been scored...

Rest of the term

This is our last meeting!

You may submit problems up until graduation...

Unless you're a senior – then you have until 5/9

Jotto... *JOHN KENNEDY* so far!

each team may need to create a valid word (easier than remembering it!)

FR
Don
Sam

Sophs

PATRICK
JANOR
CHOI
ANNA

JRs

SRs

POM-CMC-
SCR-PTZ

other

2 diner 1 diner 1
1

diner 0

diner 1

diner 1

0 bloat 3 1 bloat 1

bloat 0

bloat 1

bloat 1

0 gumbo 0 0 gumbo 0

gumbo 0

gumbo 2

gumbo 0

This term's first class to guess another's word earns 1 problem...

This term's last class to have its word guessed earns 1 problem...

Jotto so far!

each team may need to create a valid word (easier than remembering it!)

(3) [*jotto*] Not really a to-do item, but just in case it reminds folks of their hidden word new, compatible ones -- Ben H. has requested the jotto scores thus far. Here they are -

	diner	bloat	gumbo
FYears (Dan/Sam):	2	0	0
SophSelectTeam:	1	1	0
SophFunTeam:	1	3	0
JRs:	1	1	0
SRs:	0	0	0
ExtraMudd:	1	1	2
"Other" (me):	1	1	0

This term's first class to guess another's word earns 1 problem...

This term's last class to have its word guessed earns 1 problem...

Last couple of weeks... (!)

THE WALL STREET JOURNAL.  BUSINESS



Corporate Intelligence

What matters right now in business. From V world.

TECHNOLOGY MEDIA AUTOS REGULAR

5:17 pm
Feb 19, 2014 MERGERS

Facebook Values WhatsApp Users at \$42 Each

ARTICLE COMMENTS (2)

BUSINESS FACEBOOK MOBILE WHATSAPP

 Email  Print      

By TOM GARA 

According to this SEC filing, Facebook FB +0.45% is buying messaging app WhatsApp for an incredible \$19 billion dollars: \$12 billion in stock, \$4 billion in cash and another \$3 billion in restricted stock units.

At last count of the company's 55 staff, that's a cool \$345 million per employee.

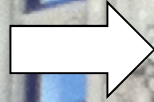
Time for
some CS
tourism!

presenting...



Stanford...

himself!?



CS tourism!



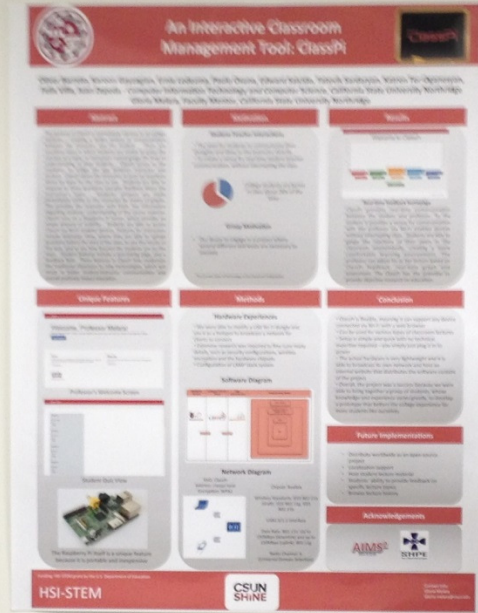


Google

Earlier conference ~ poster session...



The image shows a group of five people (three men and two women) standing in front of a poster titled "An Interactive Classroom Management Tool: ClassPi". The poster is a technical document with multiple sections including Introduction, Features, Hardware Requirements, Software Diagrams, Network Diagram, Conclusion, Future Implementations, and Acknowledgments. The people are wearing name tags and casual attire.



Earlier conference ~ poster session...

Implementation of Reagent Reducing Digital Microfluidic Biochip Algorithm

Matthew Newhill*, Philip Brisk*, Ani Sahapetian*
*California State University, Northridge
*University of California, Riverside

Laboratory-on-a-chip (LoC) automate the process of sample preparation

Synthesis Simulator Summary

UC Riverside's Digital Microfluidic Biochip Synthesis Simulator (DMBS) automates the process of sample preparation. The simulator is designed to be used by the DMBS. The simulator is designed to be used by the DMBS. The simulator is designed to be used by the DMBS.

How can we test the usefulness of this reagent re...

Integrate REMA algorithm into UC Riverside Simulator



An Interactive Classroom Management Tool: ClassPI

Oliver Barreto, Karoon Gayzagan, Ernie Ledezma, Paulo Osuna, Edward Salcido, Tatevik Sardaryan, Katren Ter-Oganesyan, Felix Villa, Juan Zepeda - Computer Information Technology and Computer Science, California State University Northridge
Gloria Melara, Faculty Mentor, California State University Northridge

Abstract

The purpose of ClassPI is to immediately obtain to all college students - creating a better setting of communication between the instructor and the student. There are countless ways in which students are unable to grasp the concept of a topic, or instructors cannot gauge the level of understanding of their students. ClassPI serves as the mediator, to bridge the gap between instructor and student. ClassPI allows the instructor to pick up questions about his topic for the class to see. Students are able to respond to these questions and give feedback about the instructor's topic. Students' answers are made immediately visible to the instructor by means of graphs. This provides the instructor with feedback, but information regarding students' understanding of the course material. ClassPI runs on a Raspberry Pi device, which provides an ample amount of security. Students are able to access ClassPI via an Android device. Features for instructors include instructor view, where they are able to upload questions before the start of the class, to see the results of the quiz, and to see live feedback that students are on the topic. Student features include a quiz-taking page, and a feedback feed. These features in ClassPI help reinforce the traditional classroom to new technologies, which will result in better student-instructor communication and overall positively impact education.

Motivation

Student Teacher Interactions

- The need for students to communicate their thoughts and ideas to the instructor directly.
- To create a venue for real-time student-teacher communication, without interrupting the class.

College students are bored in class about 28% of the time.

Group Motivation

- Our desire to engage in a project where several different skill levels are necessary to succeed.

Results

Welcome to ClassPI

Real-time feedback homepage

ClassPI provides real-time communication about between the student and professor. To the student it provides a venue for communication with the professor via their Android device without interrupting class. Students are able to upload the feedback of their peers in the classroom anonymously, creating a more comfortable learning environment. The professor can obtain his or her lecture based on ClassPI feedback, real-time graph and assessment. The ClassPI has the potential to provide adjusted research on education.

Unique Features

Welcome, Professor Melara!

Methods

Hardware Experiences

- We were able to modify a USB Wi-Fi dongle and use it as a hotspot to broadcast a network for students to connect.
- Extensive research was required to find out many details, such as security configurations, wireless encryption and the hardware chipset.
- Configuration of LEAP stack system.

Software Diagram

Network Diagram

Future Implementations

- Distribute distributable as an open-source project.
- Collaborative support.
- Share students' lecture material.
- Students' ability to provide feedback on specific lecture topics.
- Monitor network history.

Acknowledgements

HSI-STEM

CSUN SHINE

Thank you for your support and contribution to HSI-STEM.

An Interactive Classroom Management Tool: ClassPi

Zooming in...

Gayzagian, Ernie Ledezma, Paulo Osuna, Edward Salcido, Tatevik S
da - Computer Information Technology and Computer Science, California
Gloria Melara, Faculty Mentor, California State University Northridge

Motivation

Student-Teacher Interactions

- The need for students to communicate their thoughts and ideas to the instructor directly
- To create a venue for real-time student-teacher communication, without interrupting the class

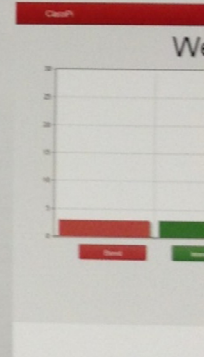


College students are bored in class about 28% of the time*.

Group Motivation

- Our desire to engage in a project where several different skill levels are necessary to succeed

*The Current State of Technology in the Classroom [Infographic]



Real-time

ClassPi provides a platform for real-time communication between the instructor and the student. It provides a venue for the professor to provide feedback without interrupting the class. The professor can gauge the real-time feedback in the classroom and provide a comfortable environment for the professor to provide feedback. ClassPi provides a platform for real-time communication between the instructor and the student. It provides a venue for the professor to provide feedback without interrupting the class. The professor can gauge the real-time feedback in the classroom and provide a comfortable environment for the professor to provide feedback.

Motivation

Student-Teacher Interactions

- The need for students to communicate their thoughts and ideas to the instructor directly
- To create a venue for real-time student-teacher communication, without interrupting the class



College students are bored
in class about 28% of the
time.

Attending posters...

I think it's the 28% of the time that the
instructor goes on irrelevant tangents about
recent travels!!



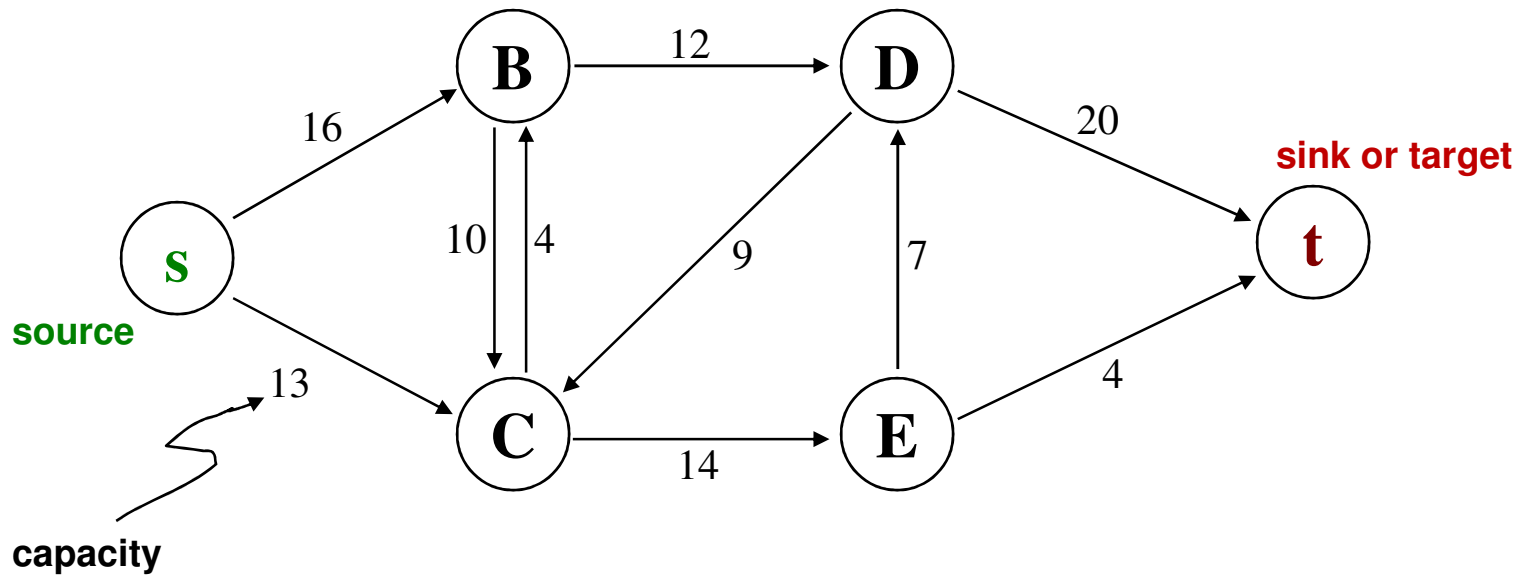
Optional open-ended project: *April*

- worth up to +8 problems ~ also, an opportunity...
- ... to try out / get familiar with / learn about a *technology, domain, library, or project*

Plan:

- (0) decide what you'd like to learn...
- (1) find a reasonable resource for it...
- (2) create a project and a write-up...
- (3) time expectation: 3 hours per week

Max Flow !



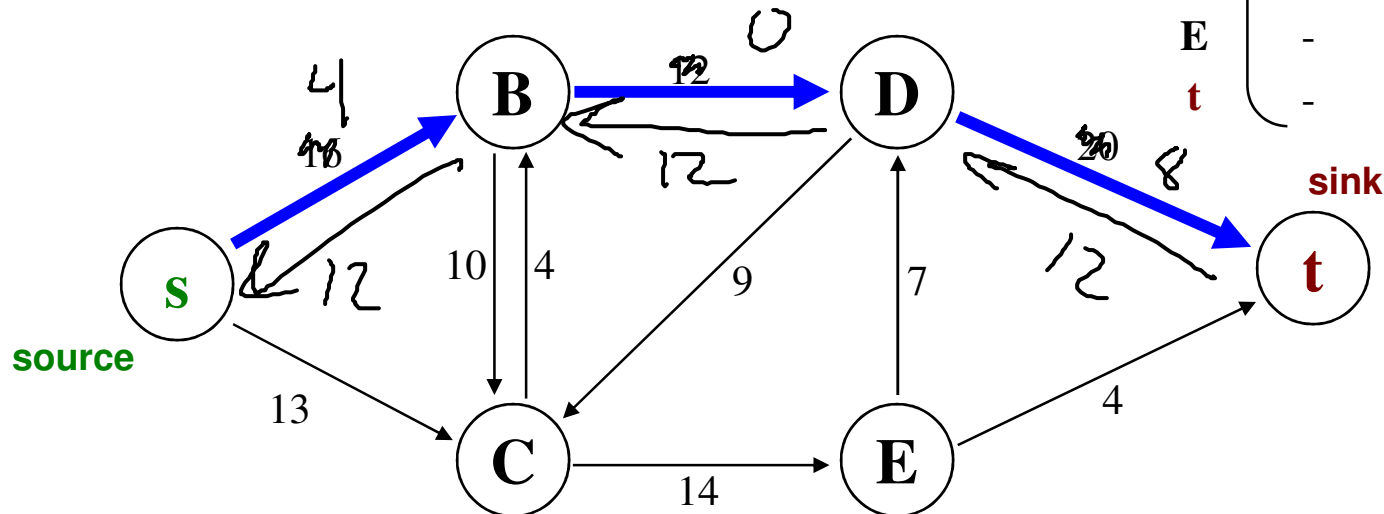
What's the maximum flow possible,
from src to sink?

Ford-Fulkerson algorithm

Max Flow

(Step #1) Use depth- or breadth-first search to find **any** path from s to t.

Capacity Graph

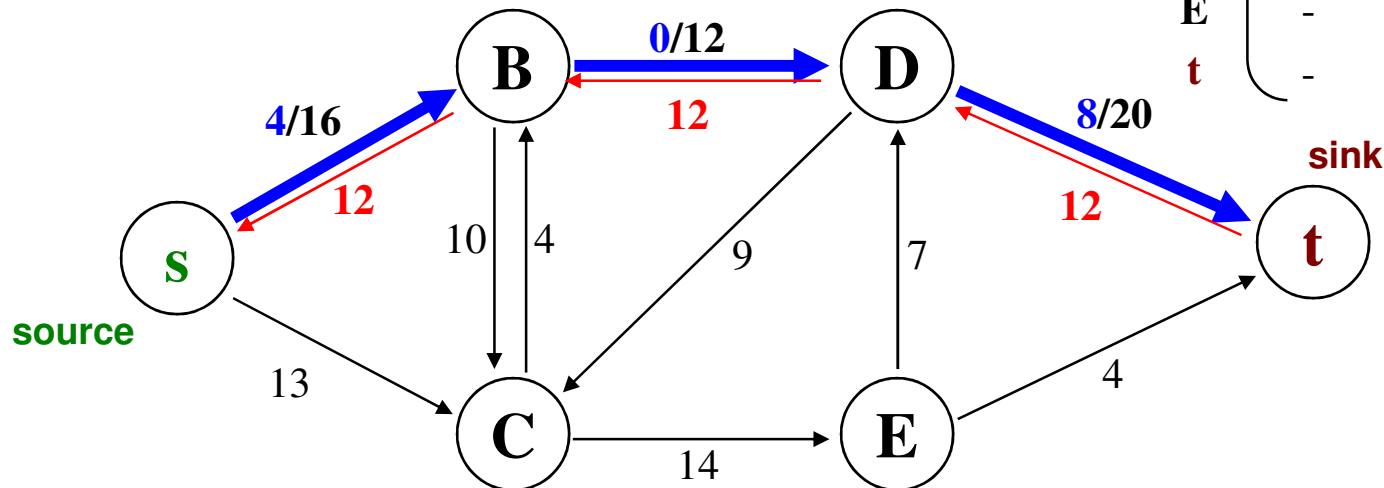


		TO					
FROM	<i>Graph</i>	s	B	C	D	E	t
	s	-	16	13	-	-	-
	B	-	-	10	12	-	-
	C	-	4	-	-	14	-
	D	-	-	9	-	-	20
	E	-	-	-	7	-	4
	t	-	-	-	-	-	-

What's left ?

Max Flow

(Step #2) Compute RESIDUAL graph



	s	B	C	D	E	t
s	-	16	13	-	-	-
B	-	-	10	12	-	-
C	-	4	-	-	14	-
D	-	-	9	-	-	20
E	-	-	-	7	-	4
t	-	-	-	-	-	-

What's left...

Residual capacities.

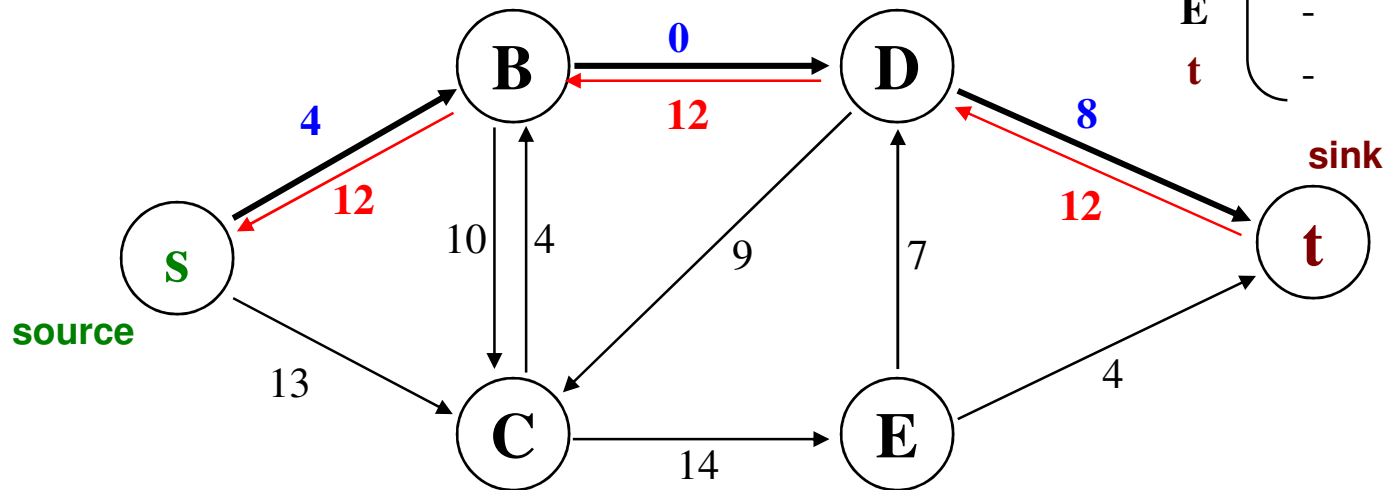
and the **red** edges?

Backwards capacities!

	s	B	C	D	E	t
s	-	4	13	-	-	-
B	12	-	10	0	-	-
C	-	4	-	-	14	-
D	-	12	9	-	-	8
E	-	-	-	7	-	4
t	-	-	-	12	-	-

Max Flow

(Step #3) Repeat until no path exists...



	s	B	C	D	E	t
s	-	16	13	-	-	-
B	-	-	10	12	-	-
C	-	4	-	-	14	-
D	-	-	9	-	-	20
E	-	-	-	7	-	4
t	-	-	-	-	-	-

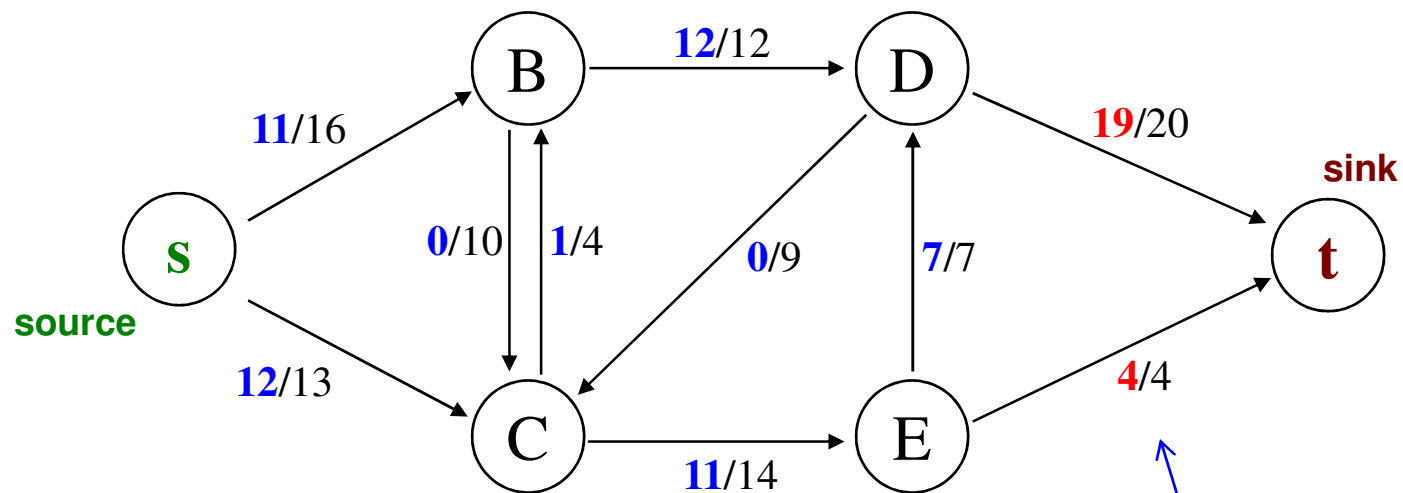
Residual capacities.

Backwards capacities.

	s	B	C	D	E	t
s	-	4	13	-	-	-
B	12	-	10	0	-	-
C	-	4	-	-	14	-
D	-	12	9	-	-	8
E	-	-	-	7	-	4
t	-	-	-	12	-	-

Max Flow

(Step #1) Use depth- or breadth-first search to find *any* path from s to t.



(Step #2) Compute RESIDUAL graph

max flow: 23

(Step #3) Repeat until no path exists...

Python...

```
if __name__ == "__main__":  
  
    # make a capacity graph  
    # node    A    B    C    D    E    F  
    C = [ [ 00, 16, 13, 00, 00, 00 ], # A  
          [ 00, 00, 10, 12, 00, 00 ], # B  
          [ 00, 04, 00, 00, 14, 00 ], # C  
          [ 00, 00,  9, 00, 00, 20 ], # D  
          [ 00, 00, 00,  7, 00,  4 ], # E  
          [ 00, 00, 00, 00, 00, 00 ] ] # F  
  
    print "C is", C  
    source = 0 # A  
    sink = 5   # F  
  
    max_flow_value = max_flow( C, source, sink )  
    print "max_flow_value is", max_flow_value
```

Linked at the ACM website by the slides...

Python...

edmonds_karp algorithm

```
def max_flow(C, source, sink):
    n = len(C) # C is the capacity matrix
    F = [[0] * n for i in range(n)] # F is the flow matrix
    # residual capacity from u to v is C[u][v] - F[u][v]

    while True:
        path = BFS(C, F, source, sink)
        if not path: break # no path - we're done!

        # find the path's flow, that is, the "bottleneck"
        edges = [C[u][v]-F[u][v] for u,v in path]
        path_flow = min( edges )

        print "Augmenting by", path_flow
        for u,v in path: # traverse path to update flow
            F[u][v] += path_flow # forward edge up
            F[v][u] -= path_flow # backward edge down

    return sum([F[source][i] for i in range(n)]) # out from source
```


Python...

```
def BFS(C, F, source, sink):
    queue = [source]          # the BFS queue
    paths = {source: []}      # stores 1 path per graph node
    while queue:

        u = queue.pop(0)      # next node to explore (expand)
        for v in range(len(C)): # for each possible next node

            # path from u to v?      and not yet at v?
            if C[u][v] - F[u][v] > 0 and v not in paths:
                paths[v] = paths[u] + [(u,v)]
                if v == sink:
                    return paths[v]

        queue.append(v)        # go from v in the future

    return None
```

Is **maxflow** good
for anything *else*?

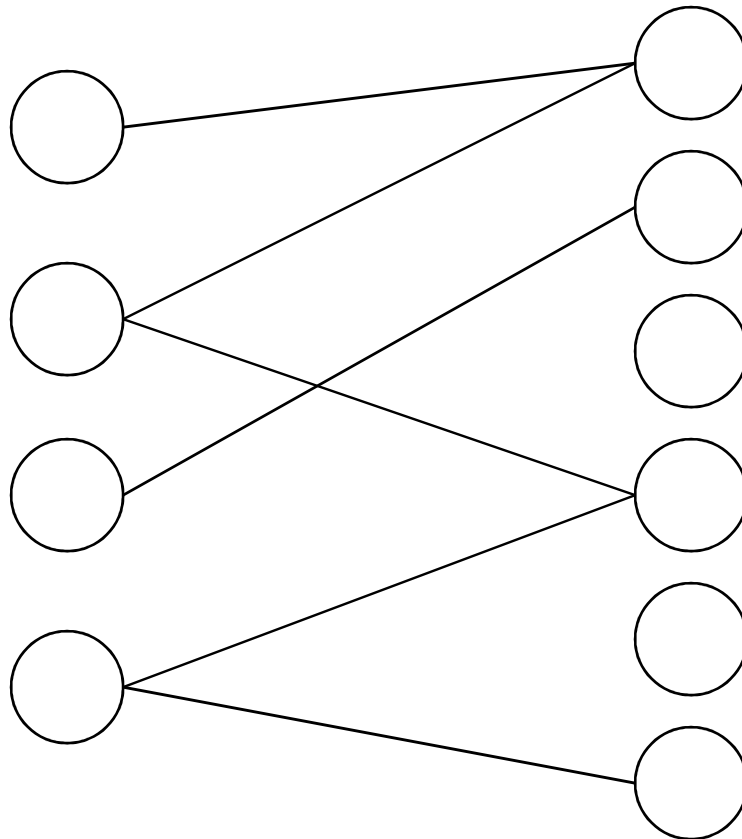
that is, beyond solving
maximum-flow problems...

Matching!

and *some* acceptable possibilities ...

we have four brides

and six grooms



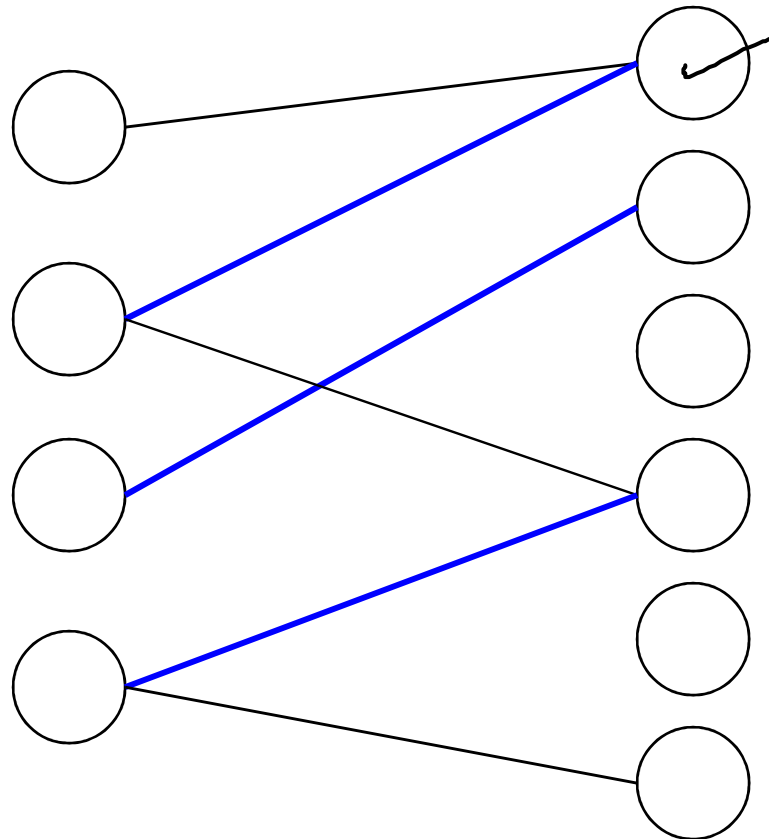
a *bipartite* graph

Matching!

and *some* acceptable possibilities ...

we have four brides

and six grooms



a *maximal* matching

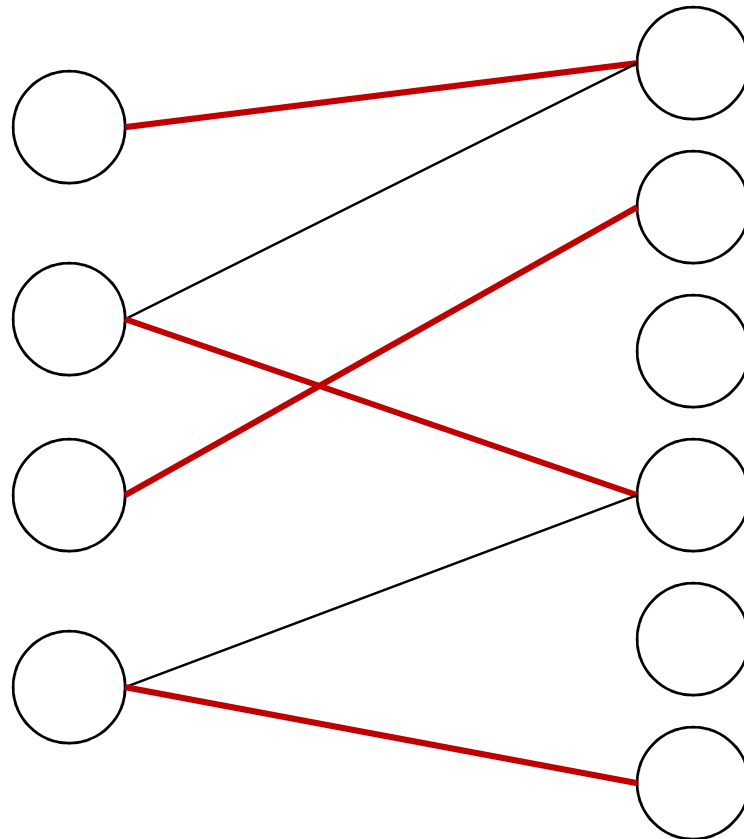
== no more matchings without rearrangement

Matching!

and *some* acceptable possibilities ...

we have four brides

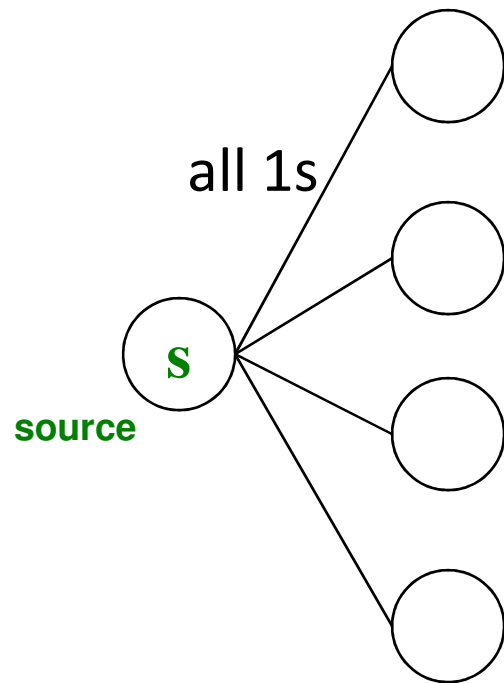
and six grooms



a *maximum* matching == no rearrangements will yield more matchings

Maximum matching *is* max flow...

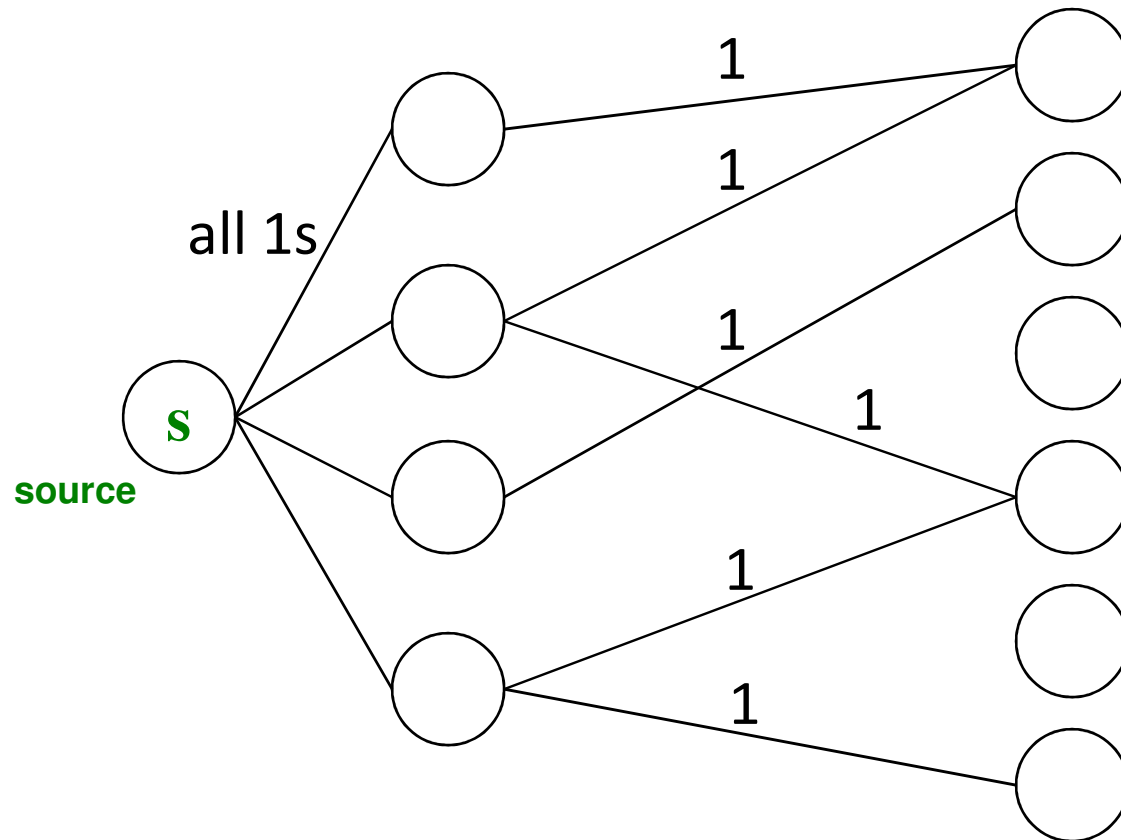
connect a source to
the left side...



Maximum matching *is* max flow...

connect a source to
the left side...

make all
capacities = 1

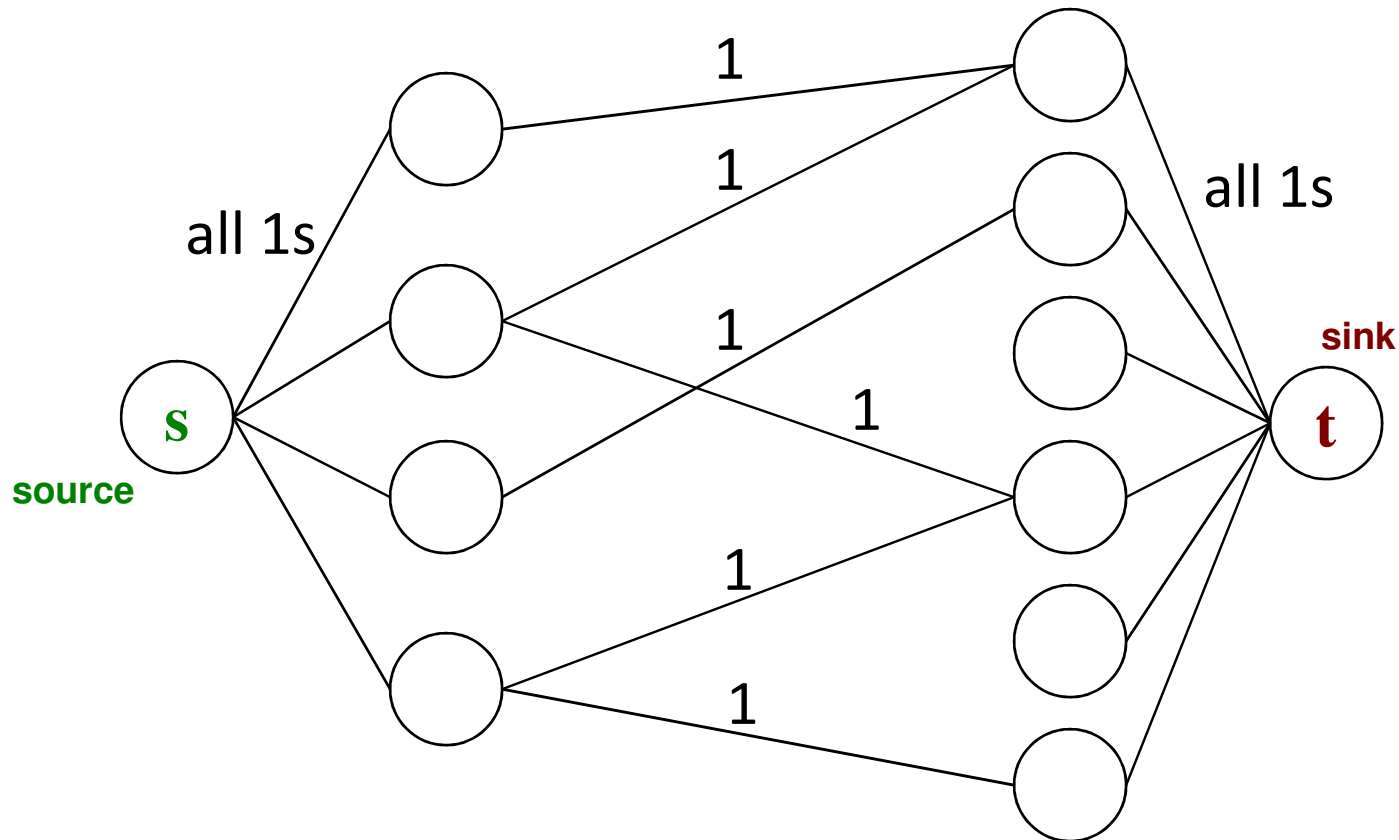


Maximum matching *is* max flow...

connect a source to
the left side...

make all
capacities = 1

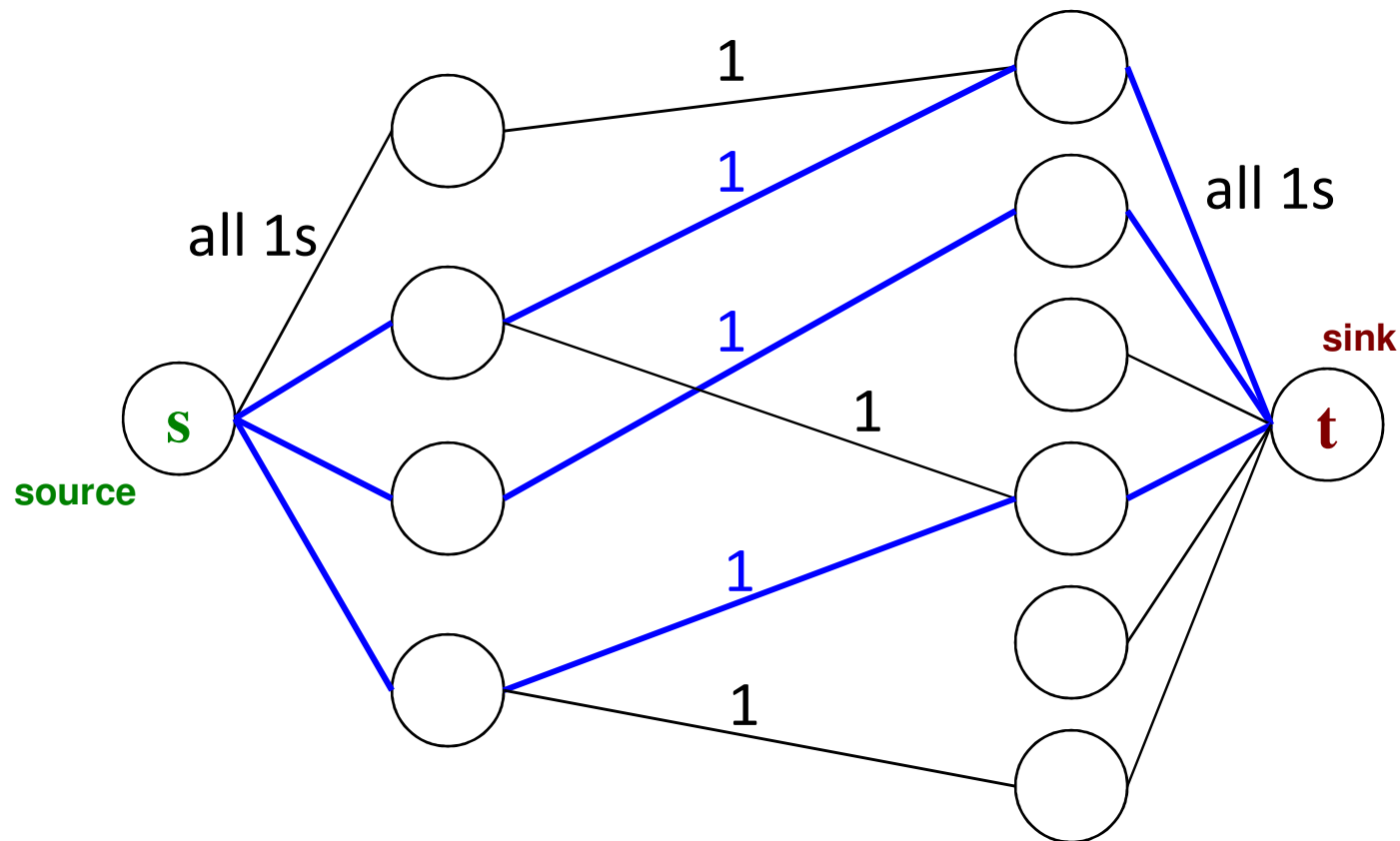
put a sink
on the right



what do the source and sink constraints ensure?

Max flow thought experiment...

Suppose this is the flow so far (3 units):

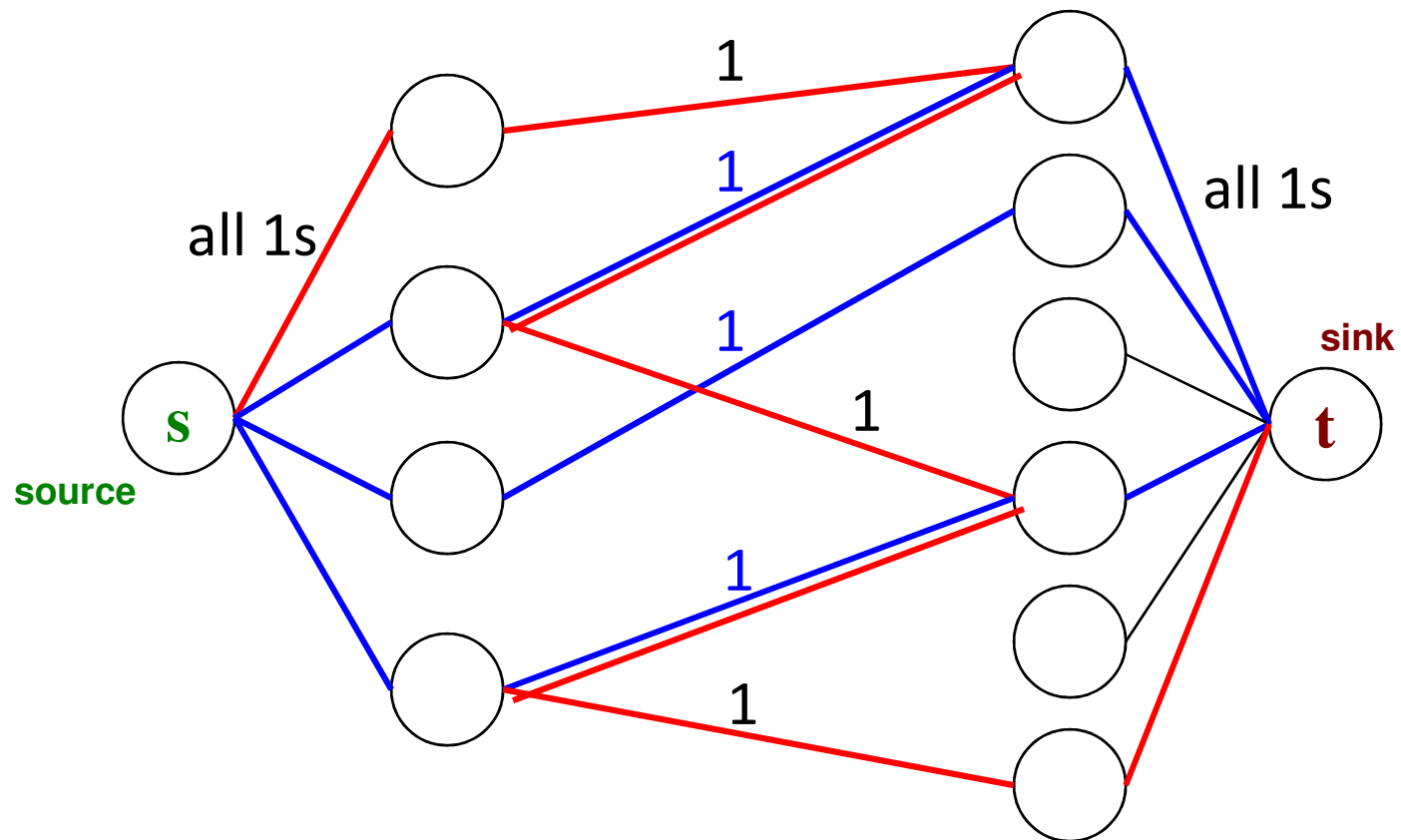


Draw what happens in the next step of the max-flow algorithm!

how to get from *maximal* matching to *maximum* matching...

Max flow thought experiment...

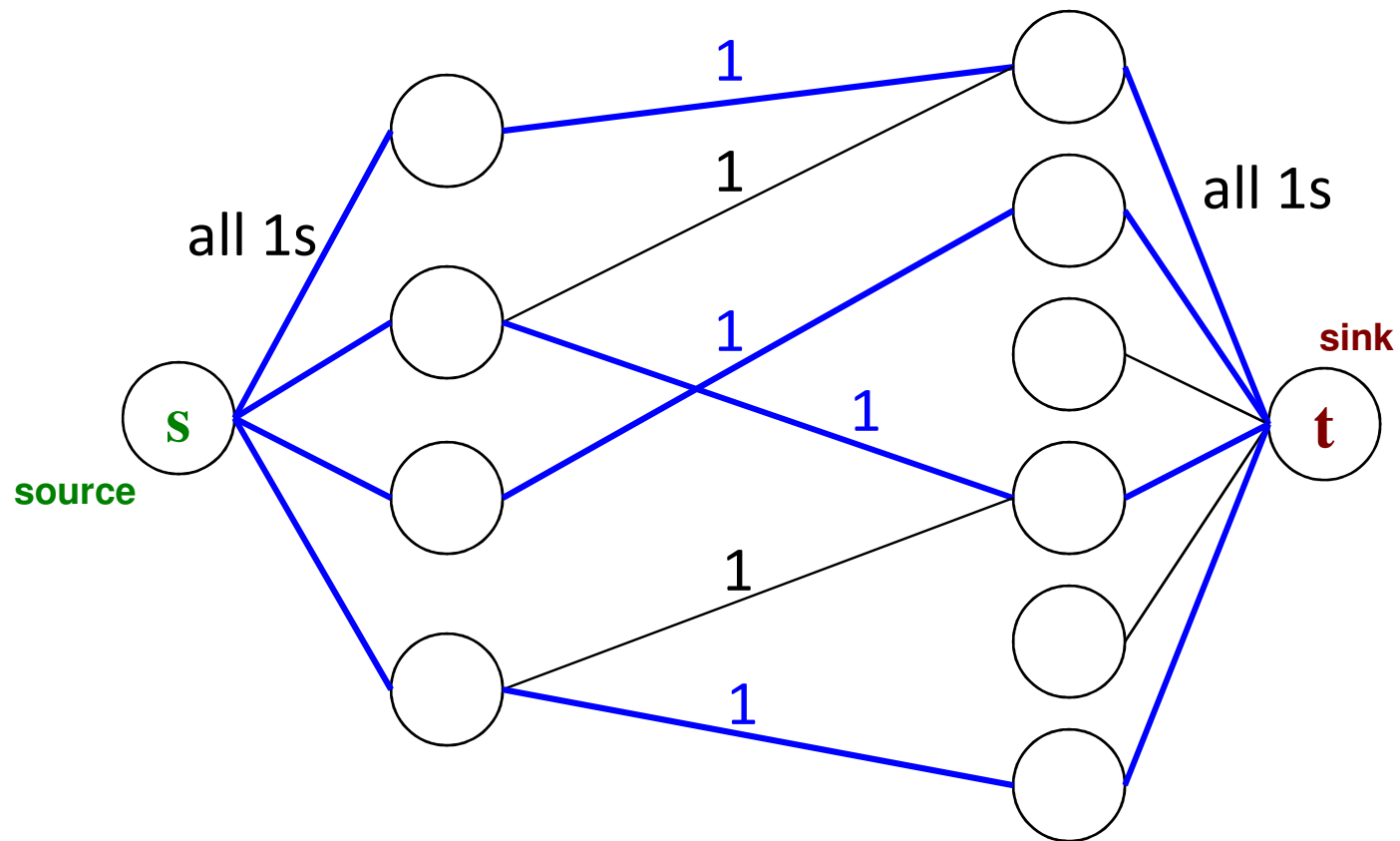
... the path it finds ...



What's going on here?

Max flow thought experiment...

Done!



Maximum matching == 4

This week's problems...

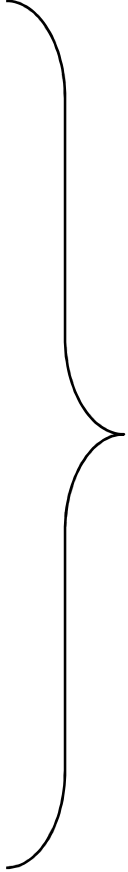
dinner

dining

hardware

muddy

feeding



all can be done
with maxflow...

The challenge:

is often *setting up* the graph

hardware

4 4 4 tools & 4 tasks

42

hammer

189

phone

10

coffee

1000

laptop

There are four
tools available ~
at these **costs**

50 1 3

20 1 3

8 2

3 1 4

each **task**
requires
some **tools**

There are four
tasks available ~
with these **rewards**

hammer

E4

phone

PHP
coding

coffee

sleep

PC

coding

Tools

Tasks

What is flowing?

How do we use the results?

hardware

4 4 4 tools & 4 tasks

42 hammer

189 phone

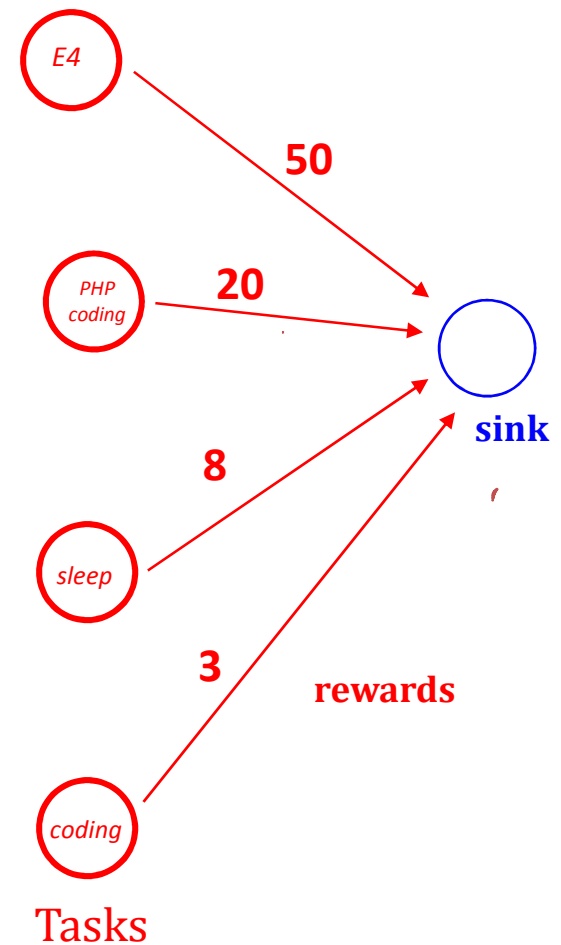
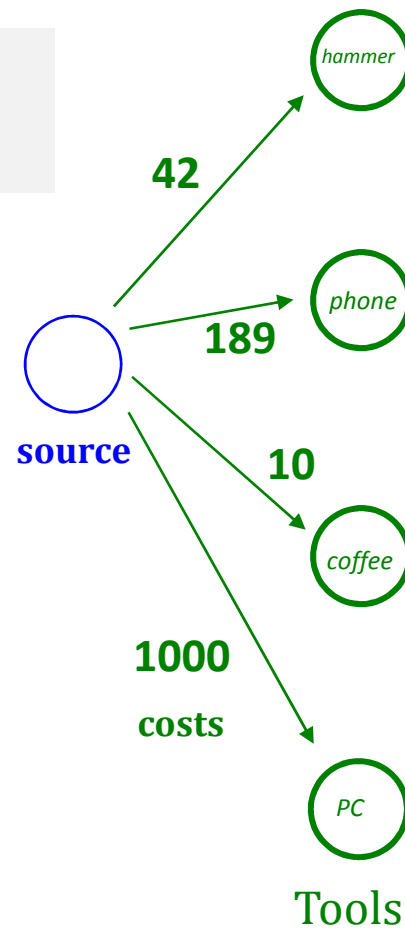
10 coffee

1000 laptop

There are four **tools** available ~ at these costs

each **task** requires some **tools**

There are four **tasks** available ~ with these rewards



What is flowing?

How do we use the results?

hardware

4 tools & 4 tasks

4 4

42

hammer

189

phone

10

coffee

1000

laptop

There are four **tools** available ~ at these costs

50 1 3

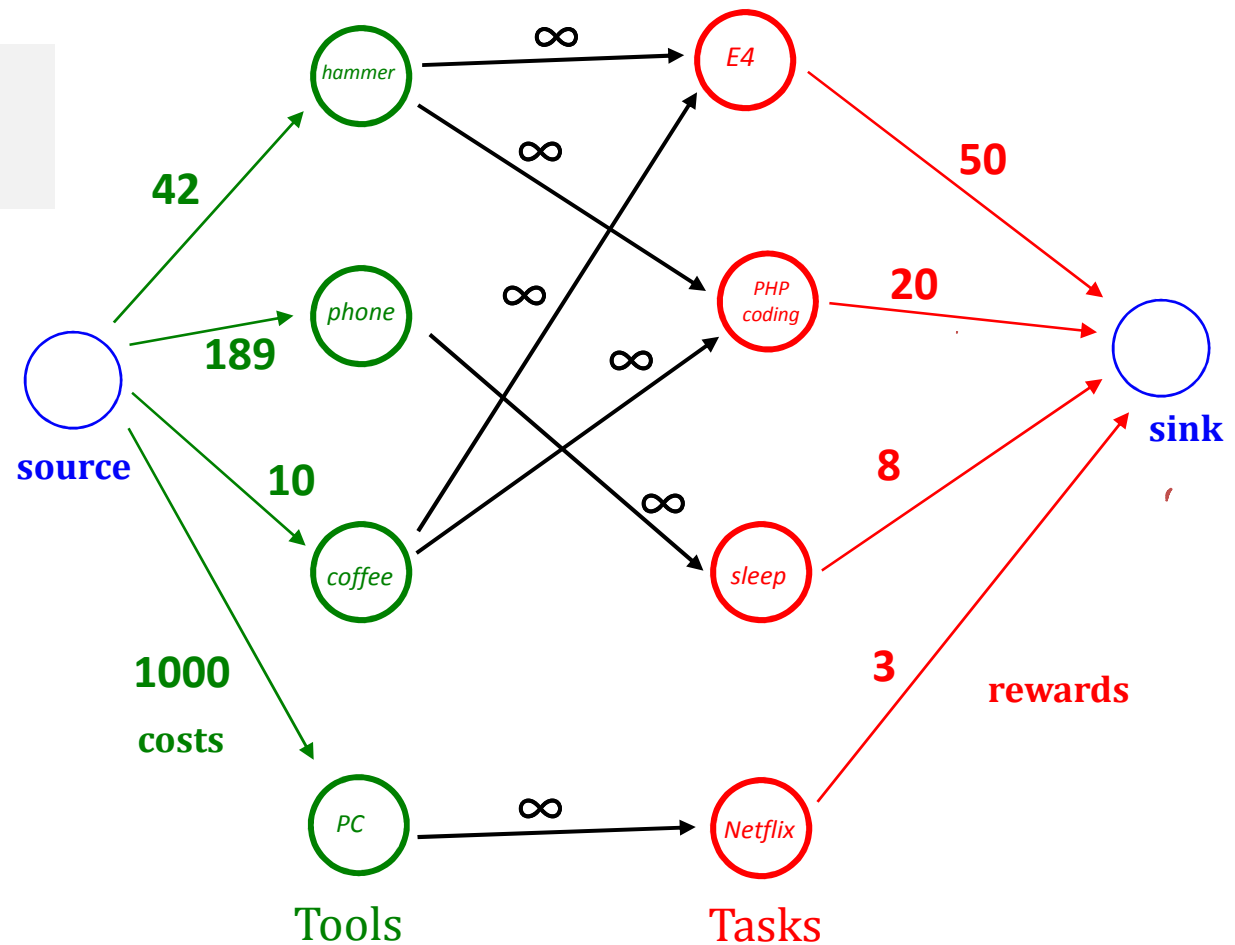
20 1 3

8 2

3 1 4

each **task** requires some **tools**

There are four **tasks** available ~ with these rewards



What is flowing?

How do we use the results?

dinner

Input

number of teams

number of tables

of people in
each team

capacity of
each table

again...

4 5
4 5 3 5
3 5 2 6 4
4 5
4 5 3 5
3 5 2 6 3
0 0

end...

Output

1
0

can an assignment be
made without putting
teammates together?

teams with sizes

4

3

5

5

seating assignments!

no teammates

tables with capacities

3

5

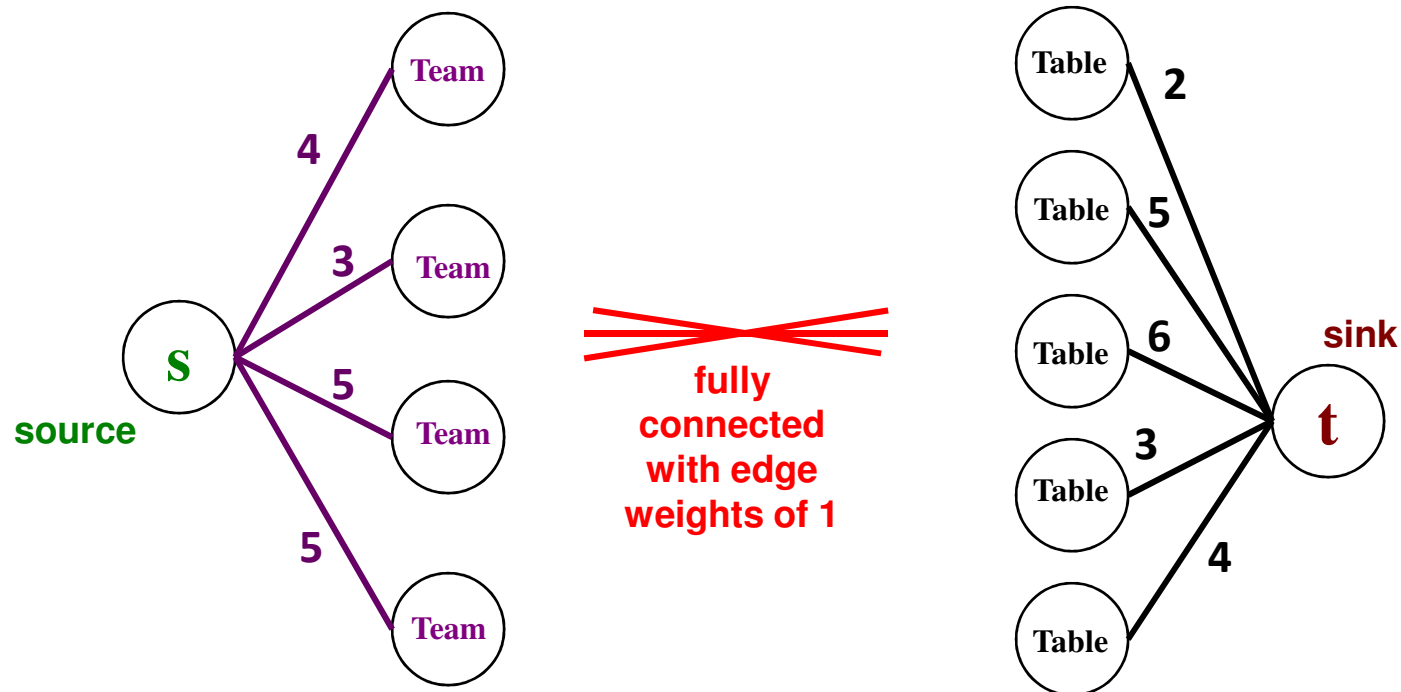
2

6

4

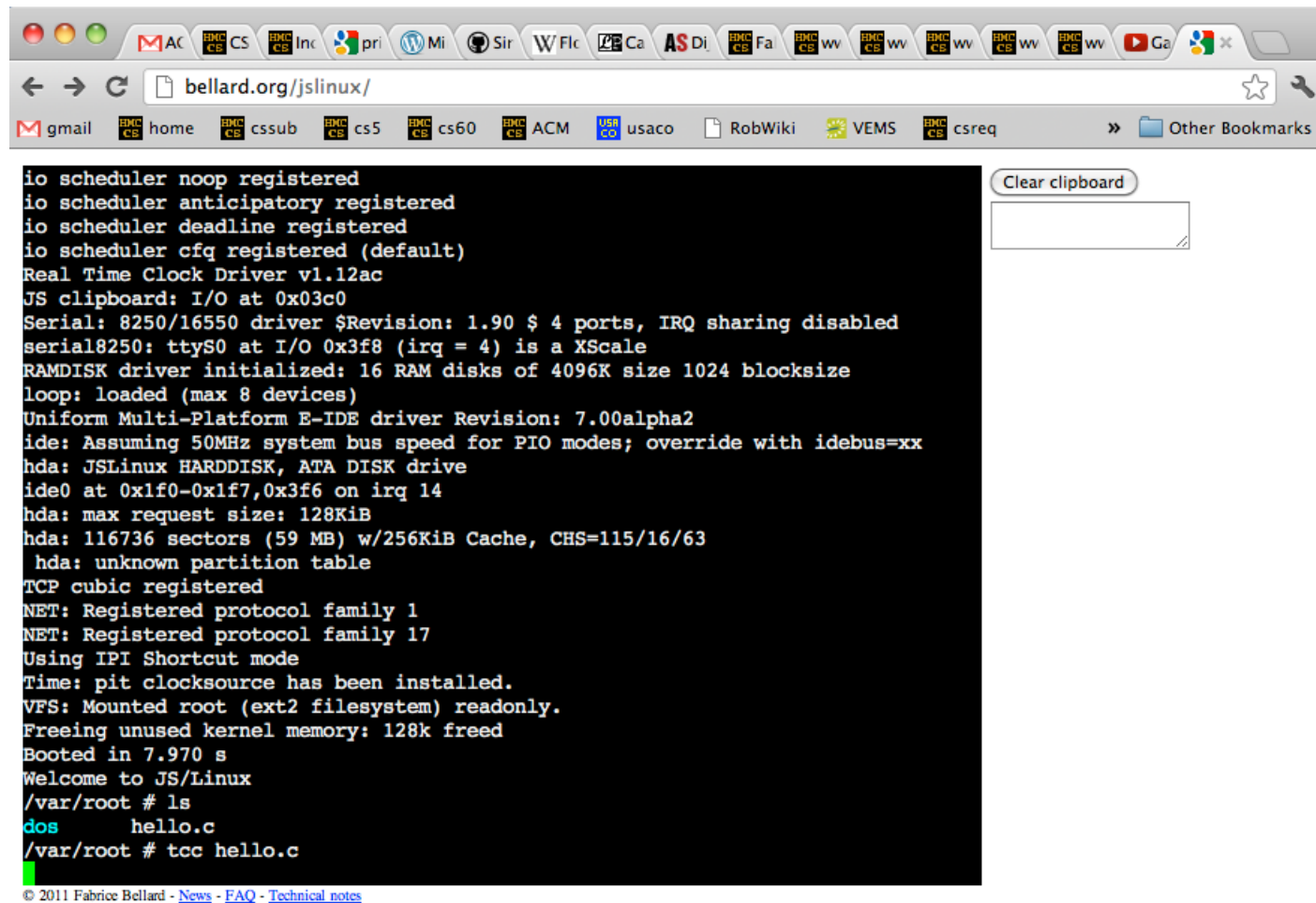
dinner

How do these edge weights reflect the problem constraints?



How does maxflow help?

What?!



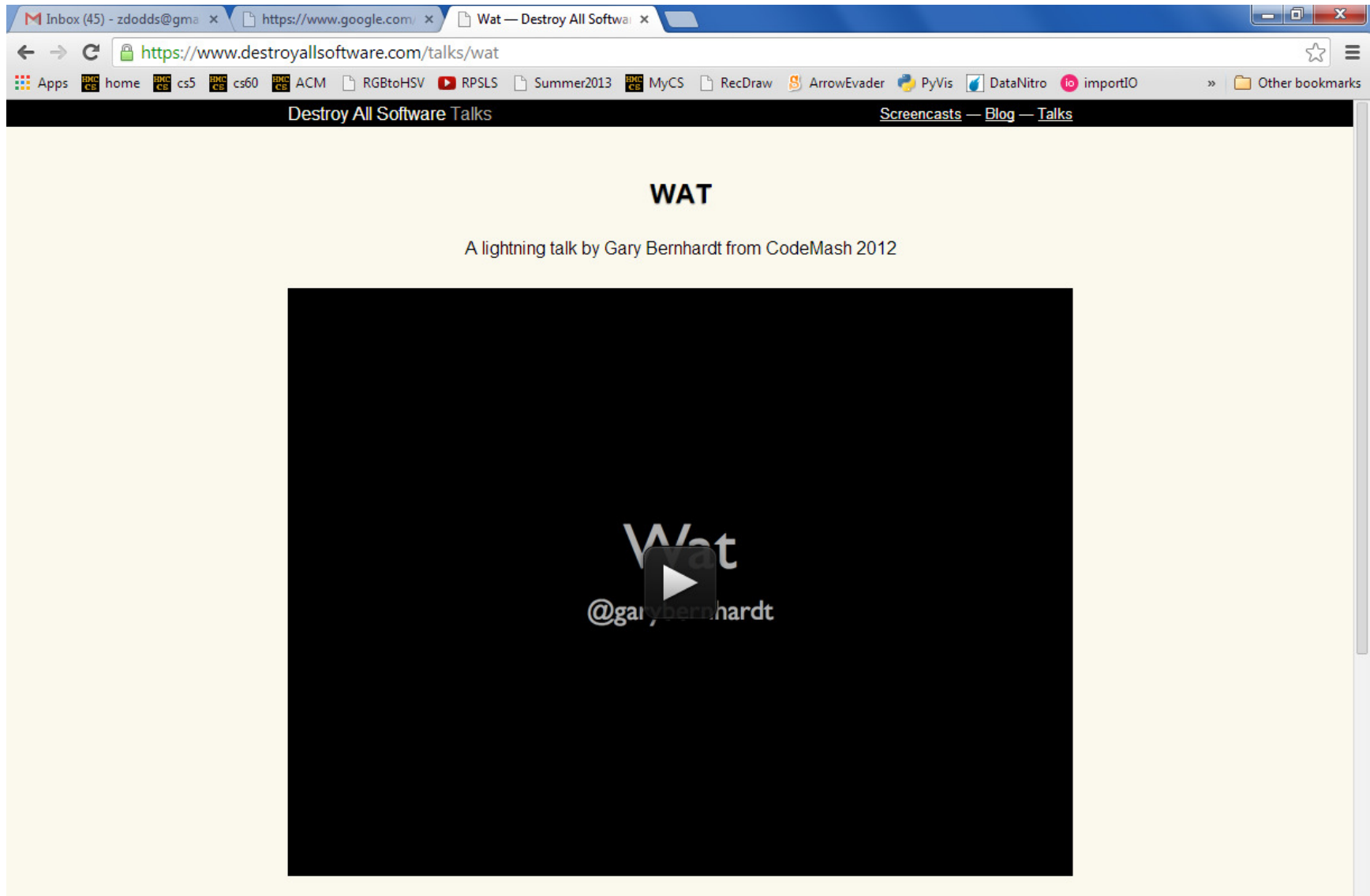
The screenshot shows a web browser window with the address bar displaying "bellard.org/jslinux/". The browser's toolbar includes various icons for email, social media, and search. Below the address bar, there are several bookmarks, including "gmail", "home", "cssub", "cs5", "cs60", "ACM", "usaco", "RobWiki", "VEMS", and "csreq". The main content area of the browser displays the output of the JSLinux boot process, which includes messages such as "io scheduler noop registered", "Real Time Clock Driver v1.12ac", "Serial: 8250/16550 driver \$Revision: 1.90 \$ 4 ports, IRQ sharing disabled", "RAMDISK driver initialized: 16 RAM disks of 4096K size 1024 blocksize", "Uniform Multi-Platform E-IDE driver Revision: 7.00alpha2", "hda: Assuming 50MHz system bus speed for PIO modes; override with idebus=xx", "hda: JSLinux HARDDISK, ATA DISK drive", "hda: max request size: 128KiB", "hda: 116736 sectors (59 MB) w/256KiB Cache, CHS=115/16/63", "hda: unknown partition table", "TCP cubic registered", "NET: Registered protocol family 1", "NET: Registered protocol family 17", "Using IPI Shortcut mode", "Time: pit clocksource has been installed.", "VFS: Mounted root (ext2 filesystem) readonly.", "Freeing unused kernel memory: 128k freed", "Booted in 7.970 s", "Welcome to JS/Linux", and a prompt for the user to enter a command. The user has entered "dos" and "hello.c", and the prompt is now "tcc hello.c".

```
io scheduler noop registered
io scheduler anticipatory registered
io scheduler deadline registered
io scheduler cfq registered (default)
Real Time Clock Driver v1.12ac
JS clipboard: I/O at 0x03c0
Serial: 8250/16550 driver $Revision: 1.90 $ 4 ports, IRQ sharing disabled
serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a XScale
RAMDISK driver initialized: 16 RAM disks of 4096K size 1024 blocksize
loop: loaded (max 8 devices)
Uniform Multi-Platform E-IDE driver Revision: 7.00alpha2
ide: Assuming 50MHz system bus speed for PIO modes; override with idebus=xx
hda: JSLinux HARDDISK, ATA DISK drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
hda: max request size: 128KiB
hda: 116736 sectors (59 MB) w/256KiB Cache, CHS=115/16/63
hda: unknown partition table
TCP cubic registered
NET: Registered protocol family 1
NET: Registered protocol family 17
Using IPI Shortcut mode
Time: pit clocksource has been installed.
VFS: Mounted root (ext2 filesystem) readonly.
Freeing unused kernel memory: 128k freed
Booted in 7.970 s
Welcome to JS/Linux
/var/root # ls
dos      hello.c
/var/root # tcc hello.c
```

© 2011 Fabrice Bellard - [News](#) - [FAQ](#) - [Technical notes](#)

Oddities from computer code...

original *Wat* talk...



php -a

```
php > $x = "209";
```

```
php > $x++;
```

```
php > print( $x ); print( "\n" );
```

210

PHP's
WAT!

```
php -a
```

```
php > $x = "209";
```

```
php > $x++;
```

```
php > print( $x ); print( "\n" );
```

```
210
```

```
php > $x = "may";
```

```
php > $x++;
```

```
php > print( $x ); print ( "\n" );
```

```
maz
```

```
php > $x++;
```

```
php > print( $x ); print ( "\n" );
```

```
mba
```

PHP's
WAT!

Now, let's try it with...

```
$x = "2d9";
```

Python's *WAT!*

WAT?

`1 == True`

`0 == False`

Are these

True

or

False

?

`(2==2) == 2`

`(1==1) == 1`

`0 == (0==0)`

`0 == (1==0)`

`1 == 1 == 1`

`0 == 0 == 0`

Good luck through
April!

Jotto guess!

April is the cruellest month...

T.S. Eliot

Jotto so far!

each team may need to create a valid word (easier than remembering it!)

(3) [*jotto*] Not really a to-do item, but just in case it reminds folks of their hidden word new, compatible ones -- Ben H. has requested the jotto scores thus far. Here they are -

	diner	bloat	gumbo
FYears (Dan/Sam):	2	0	0
SophSelectTeam:	1	1	0
SophFunTeam:	1	3	0
JRs:	1	1	0
SRs:	0	0	0
ExtraMudd:	1	1	2
"Other" (me):	1	1	0

Guesses!?

This term's first class to guess another's word earns 1 problem...

This term's last class to have its word guessed earns 1 problem...