

ORF 363/COS 323

Computing and Optimization in the Physical and Social Sciences

Amir Ali Ahmadi

Princeton, ORFE

Lecture 1

What is optimization?

- Roughly, can think of optimization as the science of making the most out of every situation.
- You've probably all done it many times this past week:

▪ What courses to take?

- To maximize learning.
- To maximize GPA (?!)
- Courses can't conflict.
- Not before 10AM.
- Professor rating > 4.5.

▪ What furniture to buy?

- To minimize cost.
- To maximize comfort.
- Must fit in your room.
- Must have 3 drawers.
- Not too heavy.

▪ Where to get dinner?

- To minimize cost.
- Less than .5 miles from dormitory.
- Must have ice cream for dessert.
- Sanitation grade > 7.

▪ Common theme:

- You make decisions and choose one of many alternatives.
- You hope to maximize or minimize something (you have an objective).
- You cannot make arbitrary decisions. Life puts constraints on you.

How is this class different from your every-day optimization?

- We'll be learning techniques for dealing with problems that have
 - Thousands (if not millions) of variables
 - Thousands (if not millions) of constraints
- These problems appear every day in the industry, in science, in engineering
- Hopeless to make decisions in your heads and with rules of thumb
- Need mathematical techniques that translate into algorithms
 - Algorithms then get implemented on a computer to solve your optimization problem
- We typically model a physical or social scenario with a precise mathematical description
- In this mathematical model, we care about actually finding *the best solution*
- Whenever we can't find the best solution, we would like to know how far off our proposed solution is

Examples of optimization problems

In finance

■ In what proportions to invest in 500 stocks?

- To maximize return.
- To minimize risk.
- No more than 1/5 of your money in any one stock.
- Transactions costs < \$70.
- Return rate > 2%.

In control engineering

■ How to drive an autonomous vehicle from A to B?

- To minimize fuel consumption.
- To minimize travel time.
- Distance to closest obstacle > 2 meters.
- Speed < 40 miles/hr.
- Path needs to be smooth (no sudden changes in direction).

In machine learning

■ How to assign likelihoods to emails being spam?

- To minimize probability of a false positive.
- To penalize overfitting on training set.
- Probability of false negative < .15.
- Misclassification error on training set < 5%.

Computing and Optimization

- This class will give you a broad introduction to
“optimization from a computational viewpoint.”
- Optimization and computing are very close areas of applied mathematics:
 - For a host of major problems in computer science, the best algorithms currently come from the theory of optimization.
 - Conversely, foundational work by computer scientists has led to a shift of focus in optimization theory from “mathematical analysis” to “computational mathematics.”
- Several basic topics in scientific computing (that we’ll cover in this course) are either special cases or fundamental ingredients of more elaborate optimization algorithms:
 - Least squares, root finding, solving linear systems, solving linear inequalities, approximation and fitting, matrix factorizations, conjugate gradients,...



Agenda for today

- Meet your teaching staff
- Get your hands dirty with algorithms
 - Game 1
 - Game 2
- Modelling with a mathematical program
 - Fermat's last theorem!
- Course logistics and expectations

Meet your teaching staff



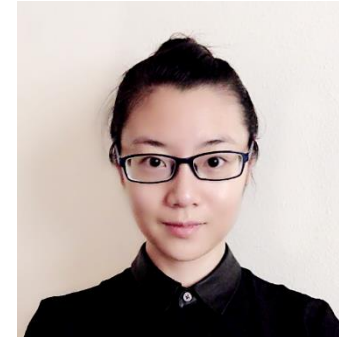
- **Amir Ali Ahmadi** (Amir Ali, or Amirali, is my first name)
 - I am a new Assistant Professor at ORFE. I come here from MIT, EECS, after a fellowship at IBM Research.
 - Office hours: **Tuesdays, 6-8 PM, Sherrerd 329.**
- <http://aaa.princeton.edu/> [a_a_a@p...](mailto:a_a_a@princeton.edu)



- **Yutong Chen** (full TA)
 - 3rd year grad at PACM
 - Office hours: **Mon 6-8, Sherrerd 003**
- [yutong@math.p...](mailto:yutong@math.princeton.edu)



- **Georgina Hall*** (1/2 TA)
 - 3rd year grad at ORFE
 - Office hours: **Wed 5-7, Sherrerd 107**
- [gh4@p...](mailto:gh4@princeton.edu)

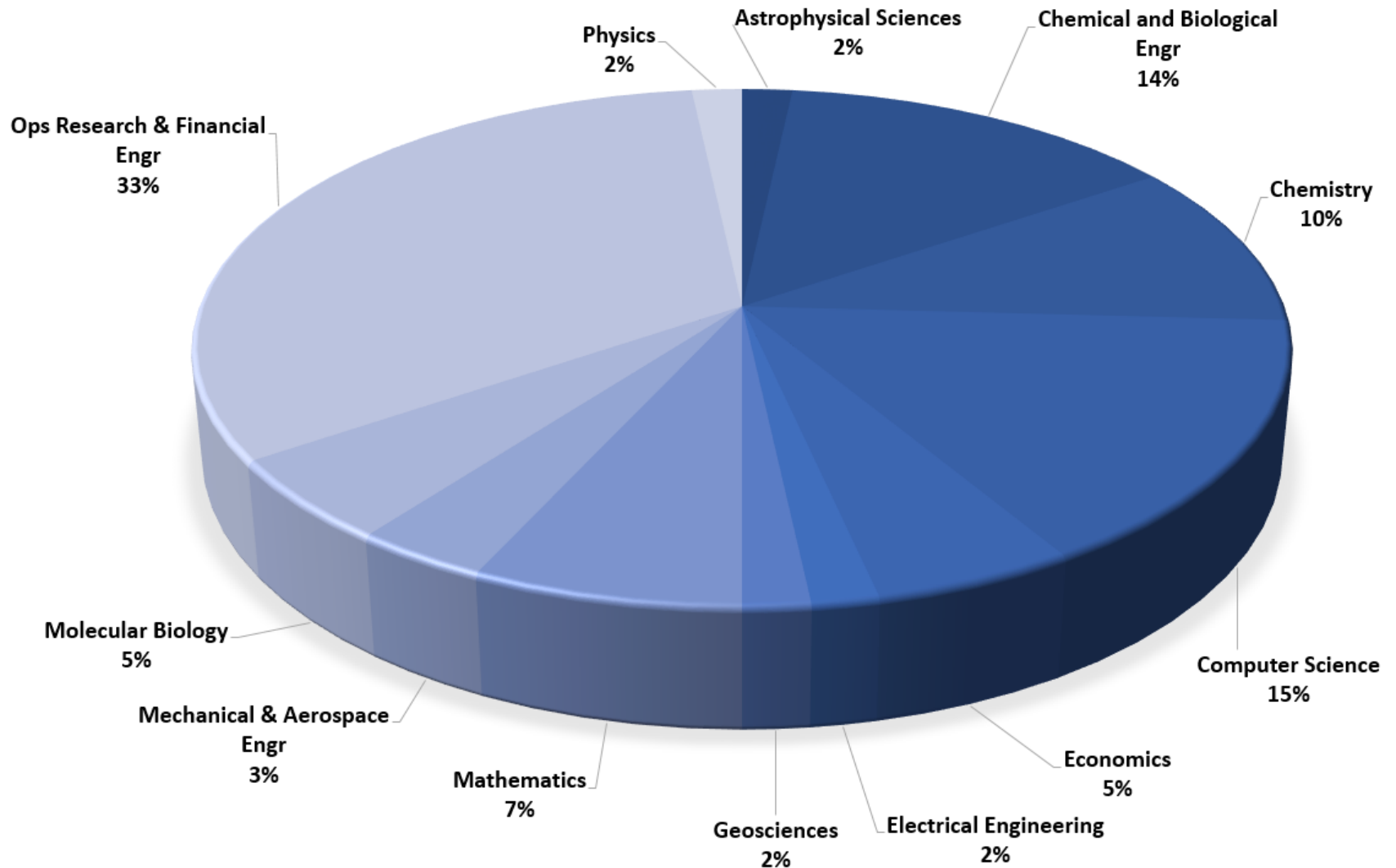


- **Jing Ye** (1/2 TA)
 - 2rd year grad at ORFE
 - Office hours: **Th 5-7, Sherrerd 107**
- [jingy@p...](mailto:jingy@princeton.edu)

*Georgina has helped a lot with course development! We gratefully acknowledge the support of the Keller Center and ORFE.

Meet your classmates

ORF 363/COS 323, Fall 2014 (57 students)



Let's get to the games!

Let's ship some oil together!

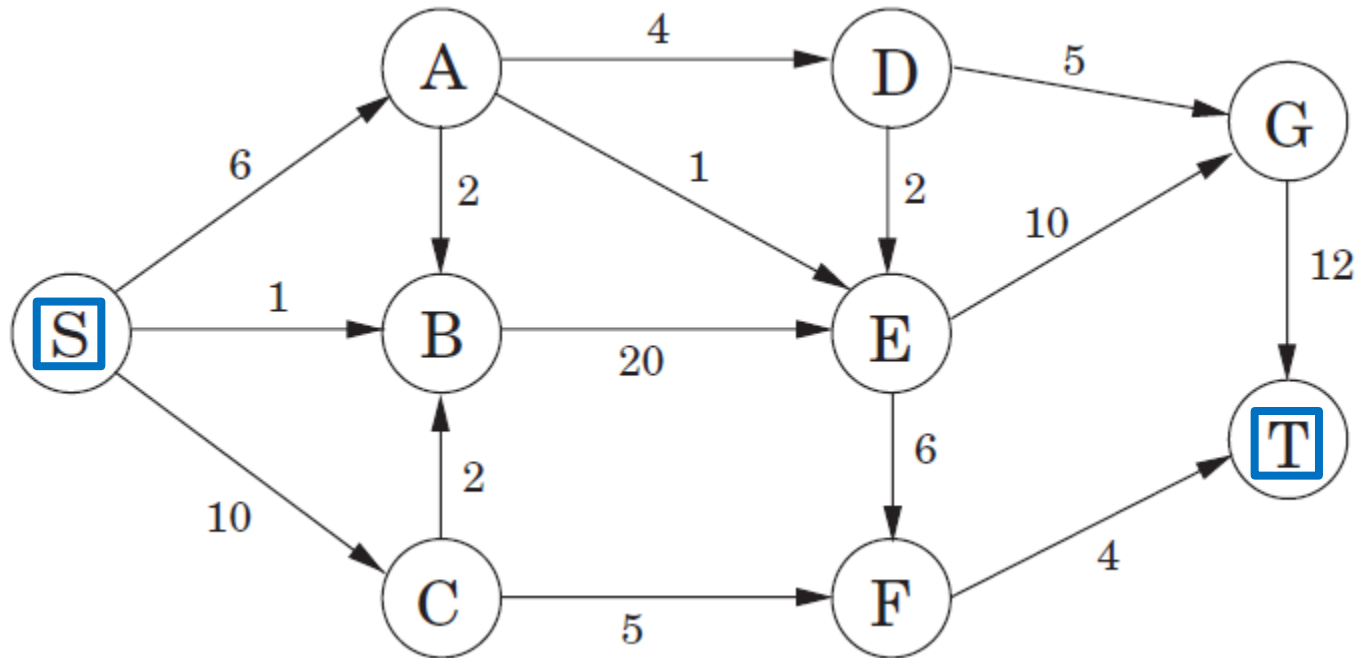
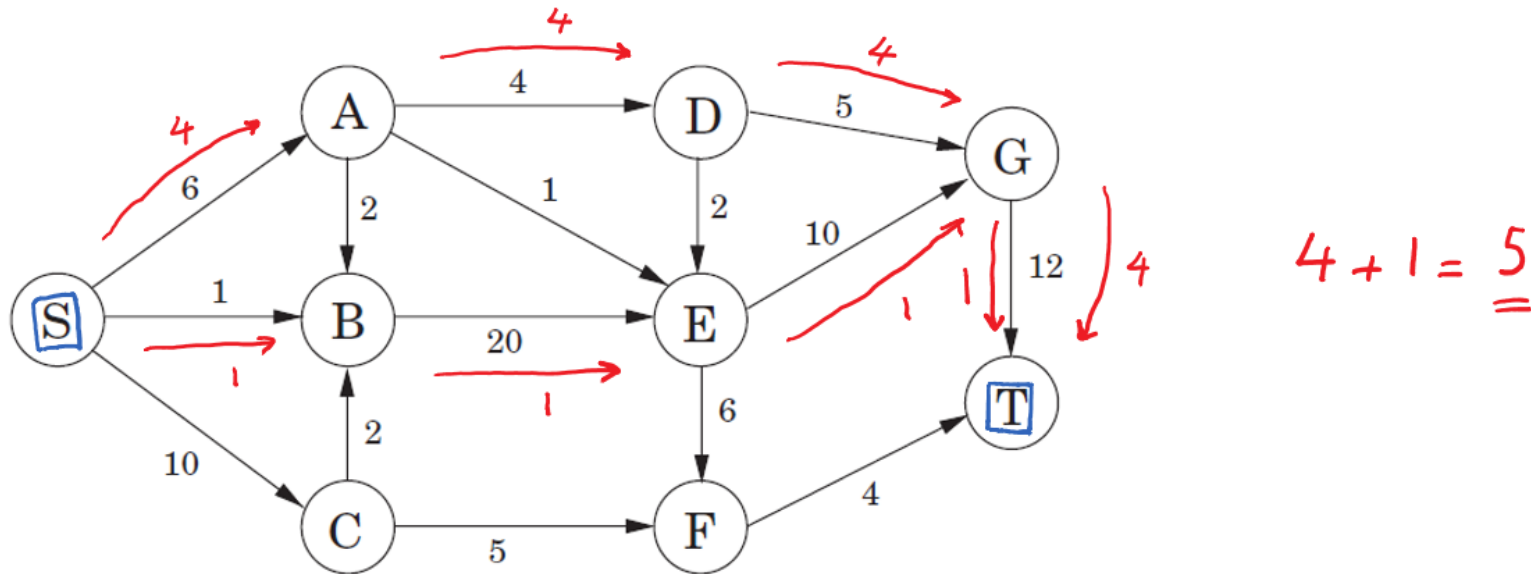


Image credit: [DPV08]

■ Rules of the game:

- Cannot exceed capacity on the edges.
- For each node, except for S and T, flow in = flow out (i.e., no storage).
- **Goal:** ship as much oil as you can from S to T.

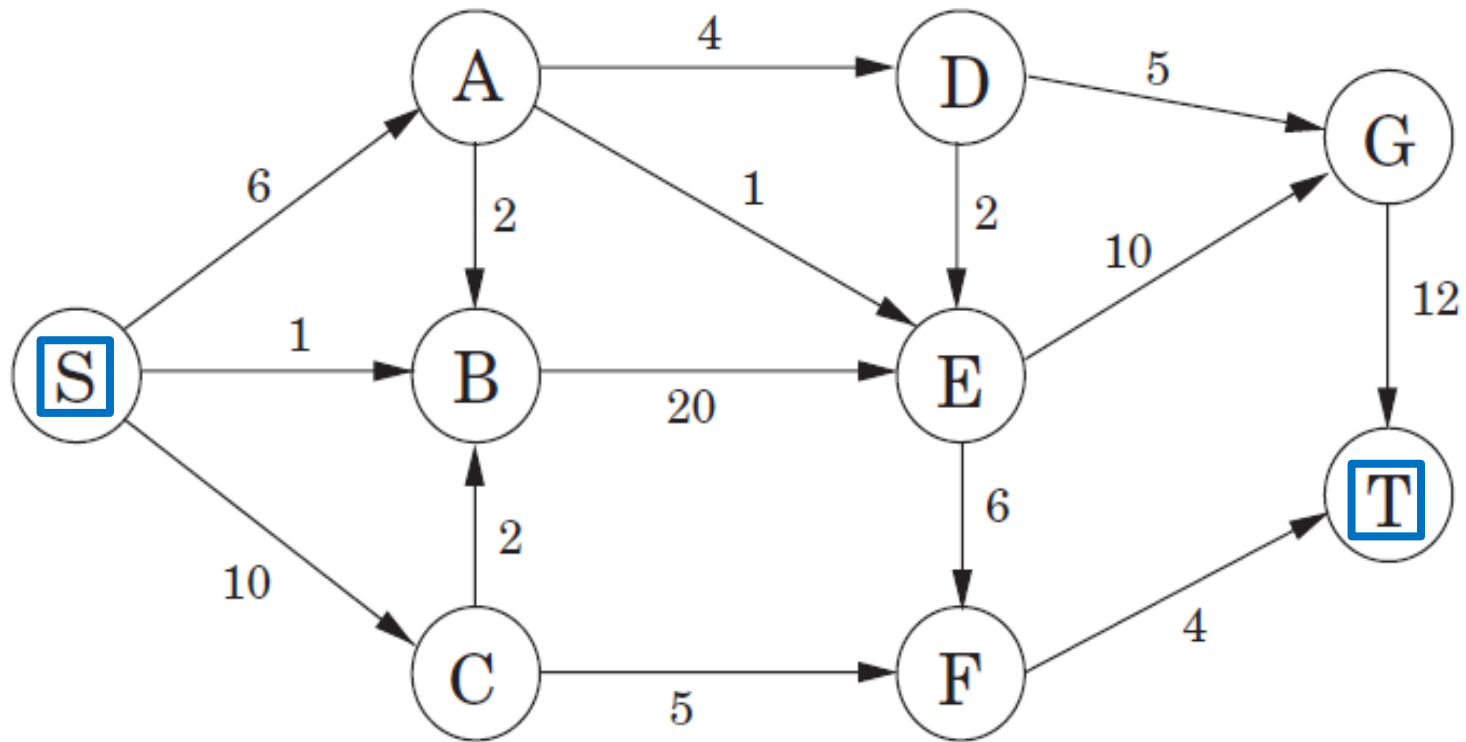
- Let me start things off for you. Here is a flow with value 5:



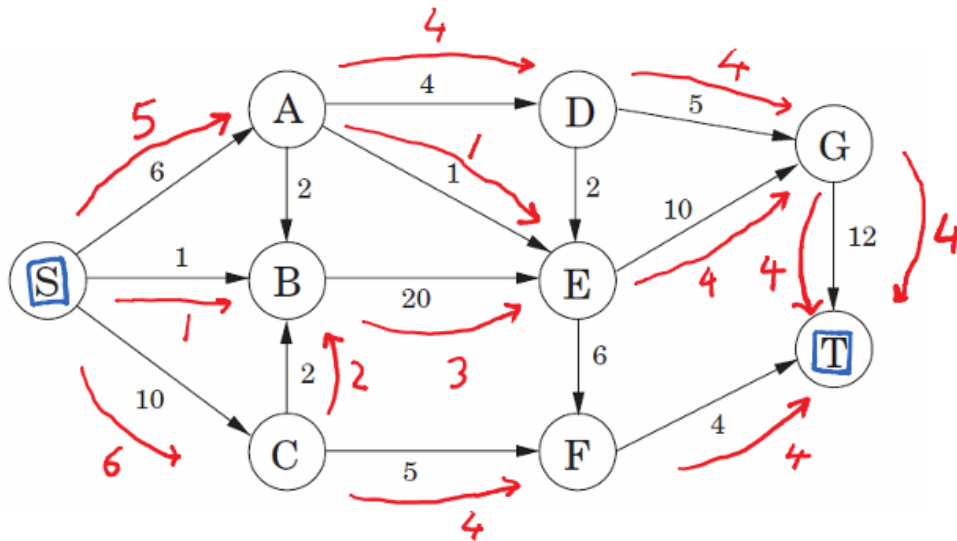
- Can you do better?
- How much better?
- You all get a copy of this graph on the handout.

- You have 5 minutes!

You tell me, I draw...

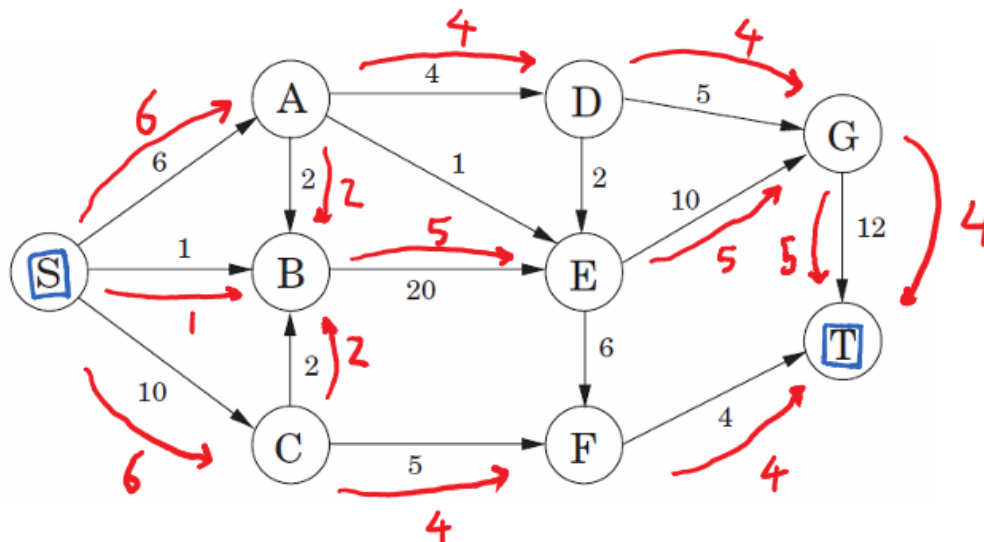


A couple of good attempts



$$4 + 4 + 4 = \underline{\underline{12}}$$

- Can you do better?

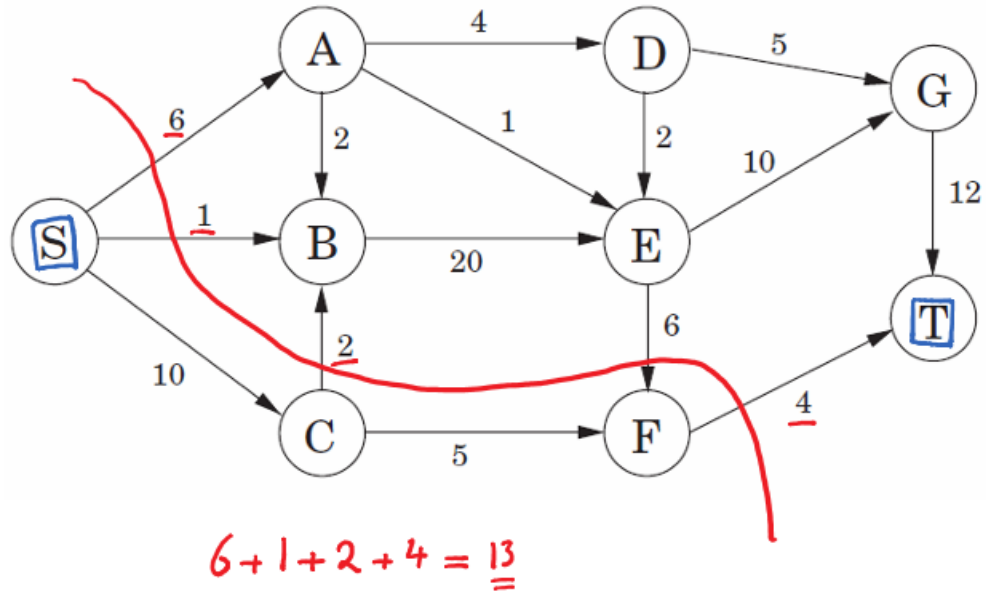


$$5 + 4 + 4 = \underline{\underline{13}}$$

- Can you do better?
- How can you convince someone that it's impossible to do better?

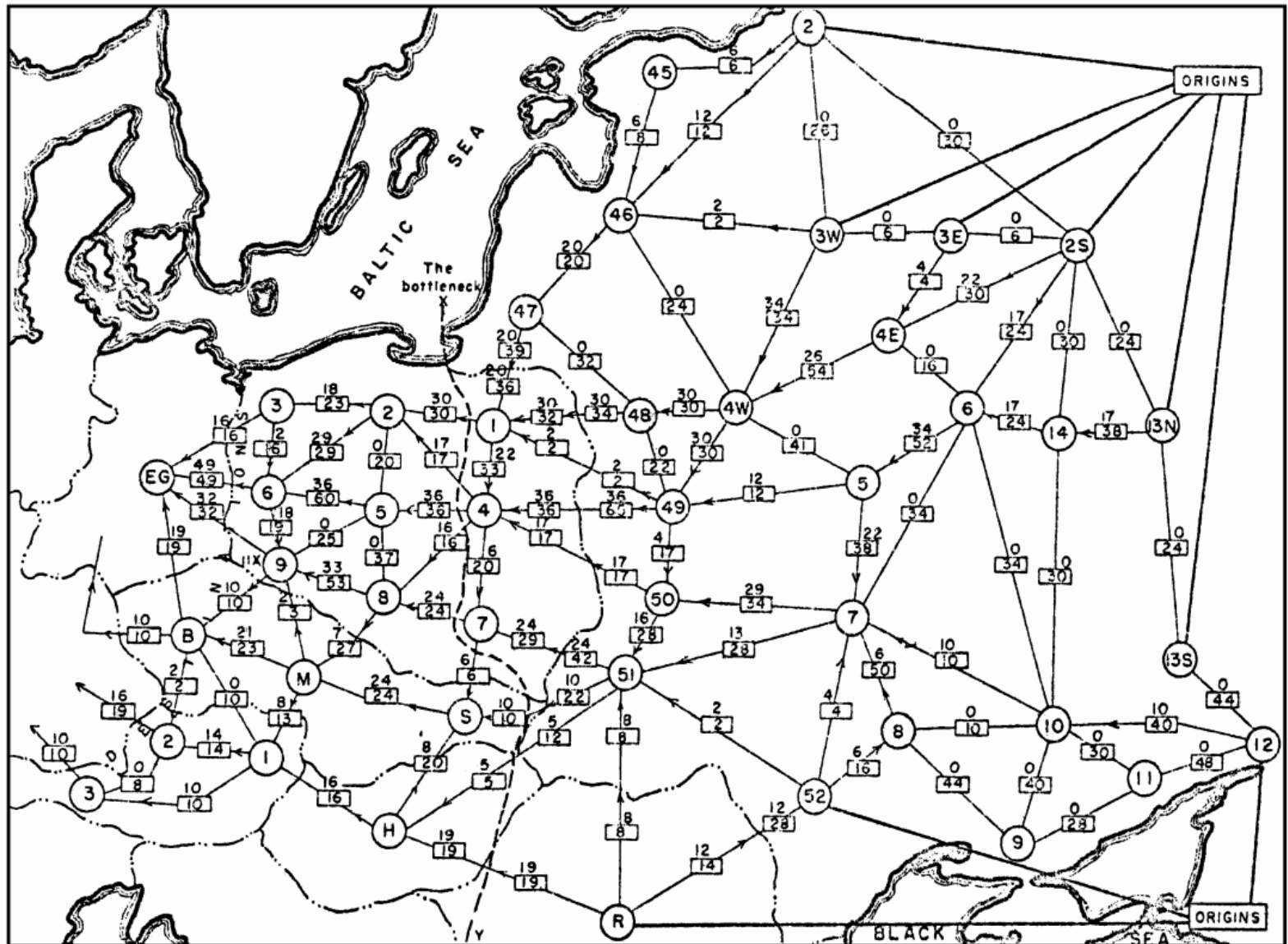
13 is the best possible!

- Proof by magic:



- The rabbit is the red “cut”!
- Any flow from S to T must cross the red curve.
- So it can have value at most 13.
- And here is the magic: such a proof is *always* possible!

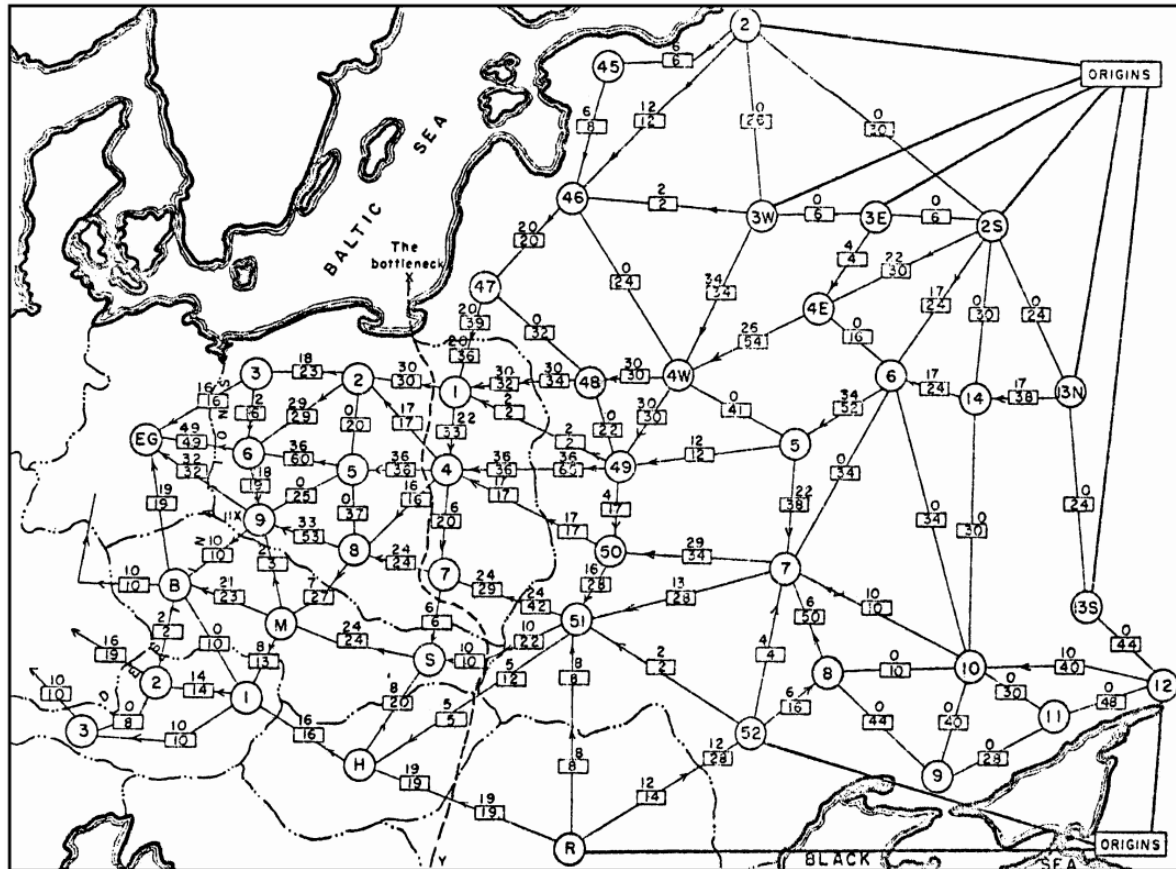
Let's try a more realistic graph



■ How long do you think an optimization solver would take (on my laptop) to find the best solution here?

■ How many lines of code do you think you have to write for it?

■ How would someone who hasn't seen optimization approach this?



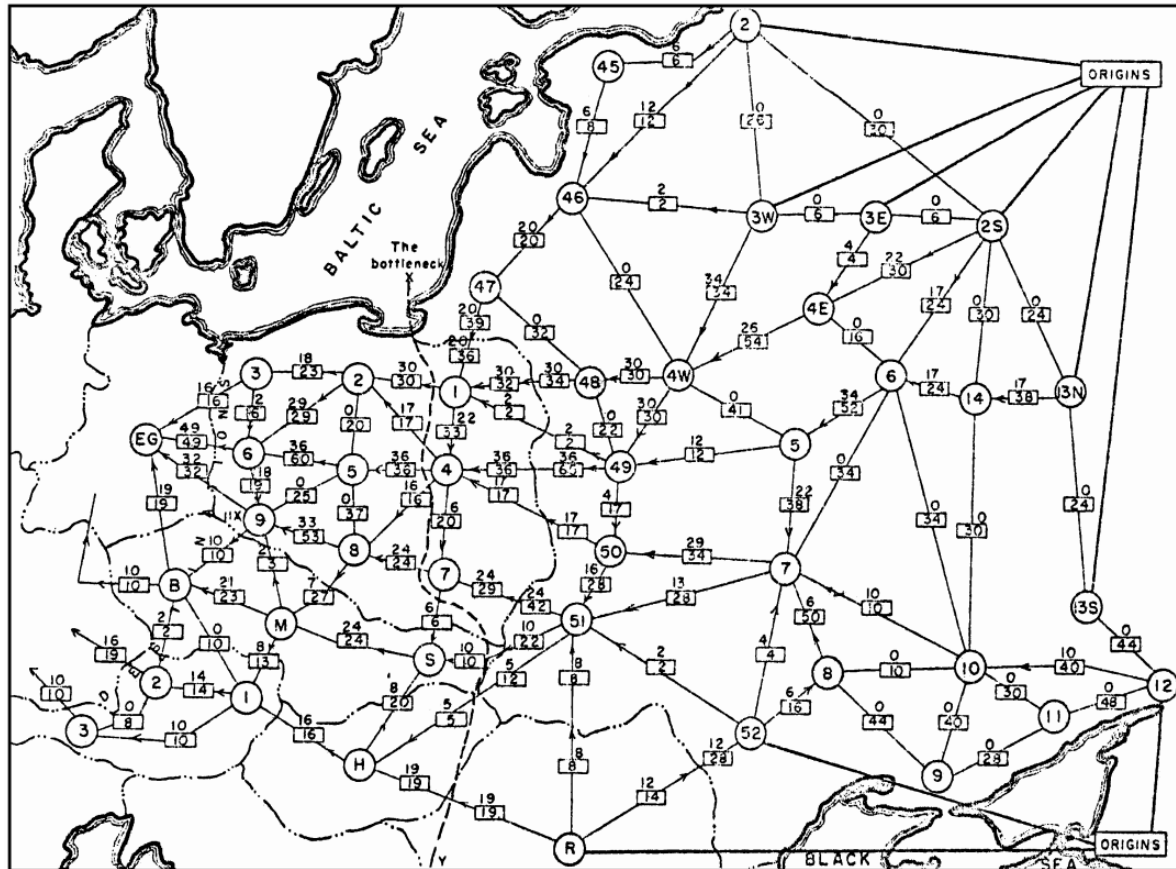
■ Trial and error?

■ Push a little flow here, a little there...

■ Do you think they are likely to find the best solution?

A bit of history behind this map

- From a secret report by Harris and Ross (1955) written for the Air Force.
- Railway network of the Western Soviet Union going to Eastern Europe.
- Declassified in 1999.
- Look at the min-cut on the map (called the “bottleneck”)!
- There are 44 vertices, 105 edges, and the max flow is 163K.



- Harris and Ross gave a heuristic which happened to solve the problem optimally in this case.
- Later that year (1955), the famous Ford-Fulkerson algorithm came out of the RAND corporation. The algorithm always finds the best solution (for rational edge costs).

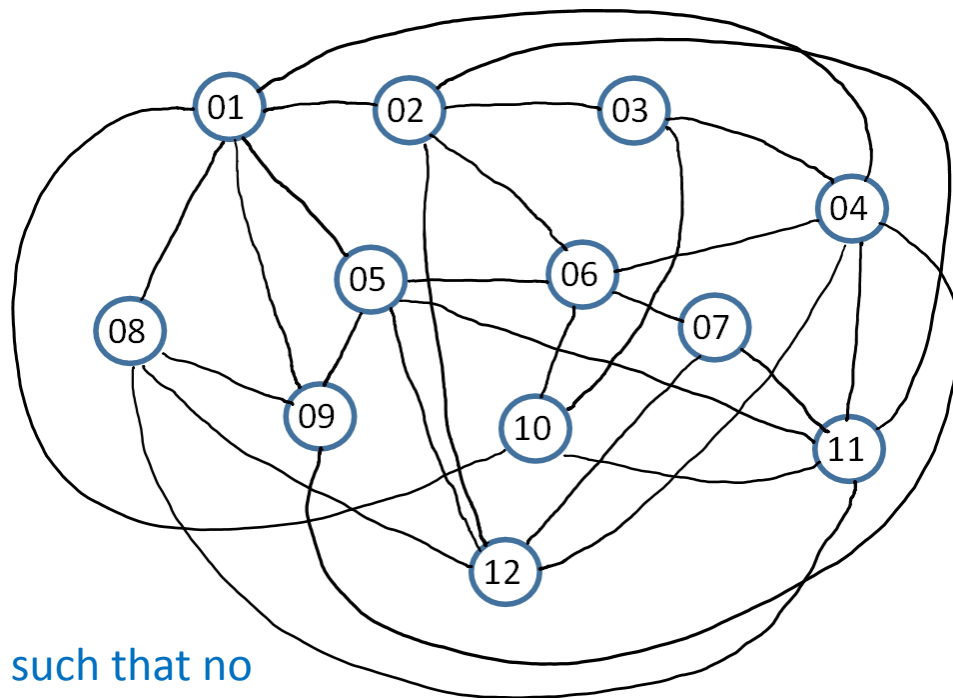
Let's look at a second problem

...and tell me which one you
thought was easier

Two finals in one day? No thanks.

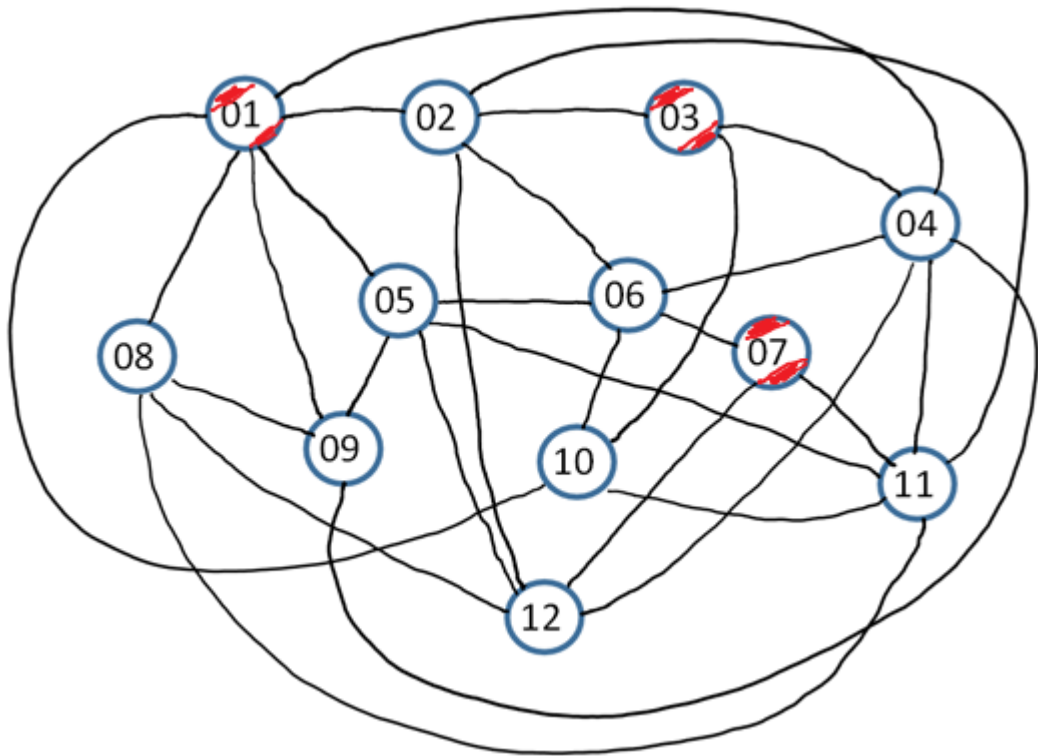
- The department chair at ORFE would like to schedule the final exams for 12 graduate courses offered this semester.
- He wants to have as many exams as possible on the same day, so everyone gets done quickly and goes on vacation.
- There is just one constraint:
No student should have >1 exam on the same day.

- The nodes of this graph are the 12 courses.
- There is an edge between two nodes if (and only if) there is at least one student who is taking both courses.
- If we want to schedule as many exams as possible on the same day, what are we looking for in this graph?



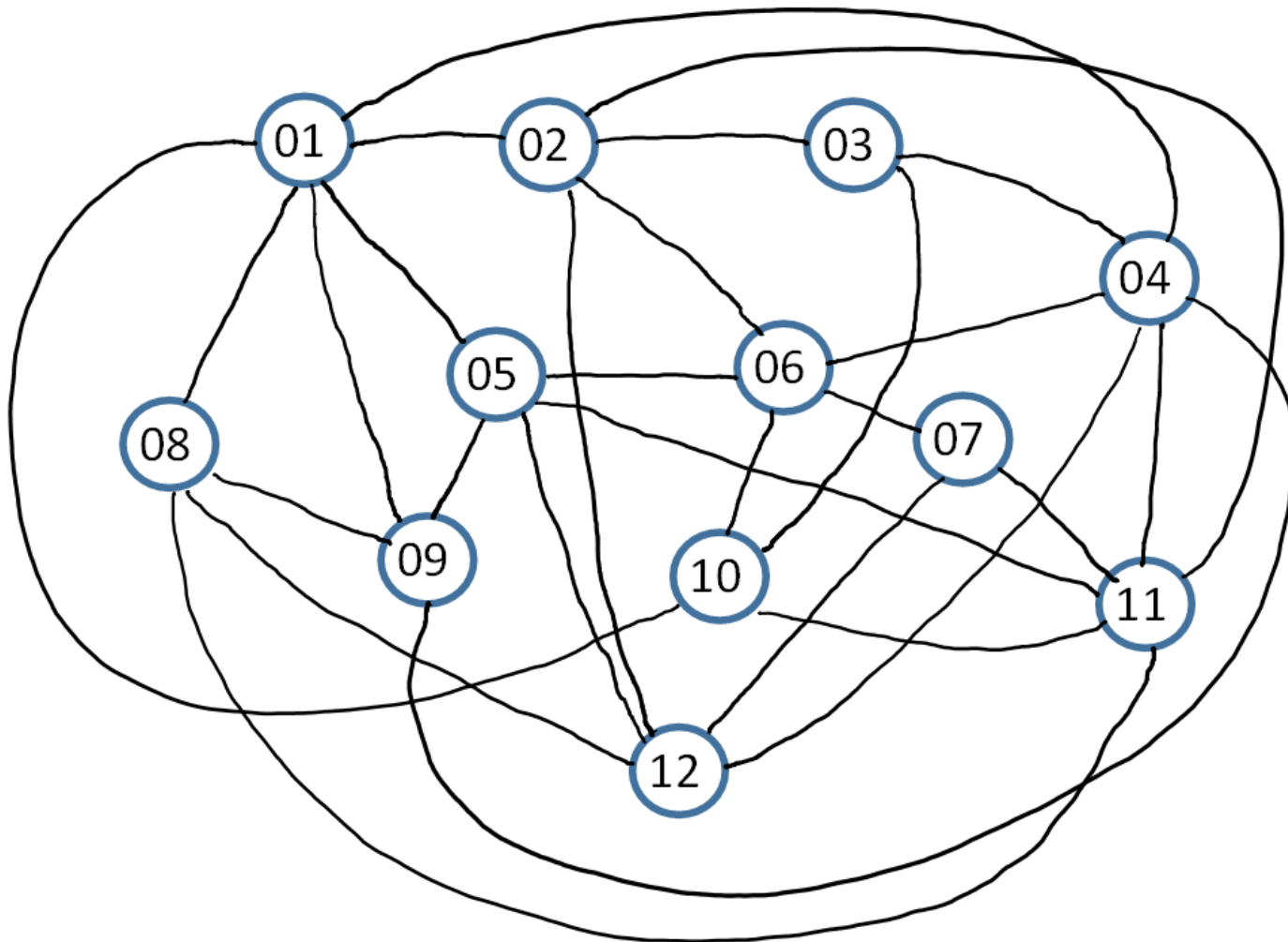
- The largest collection of nodes such that no two nodes share an edge.

■ Let me start things off for you. Here is 3 concurrent final exams:

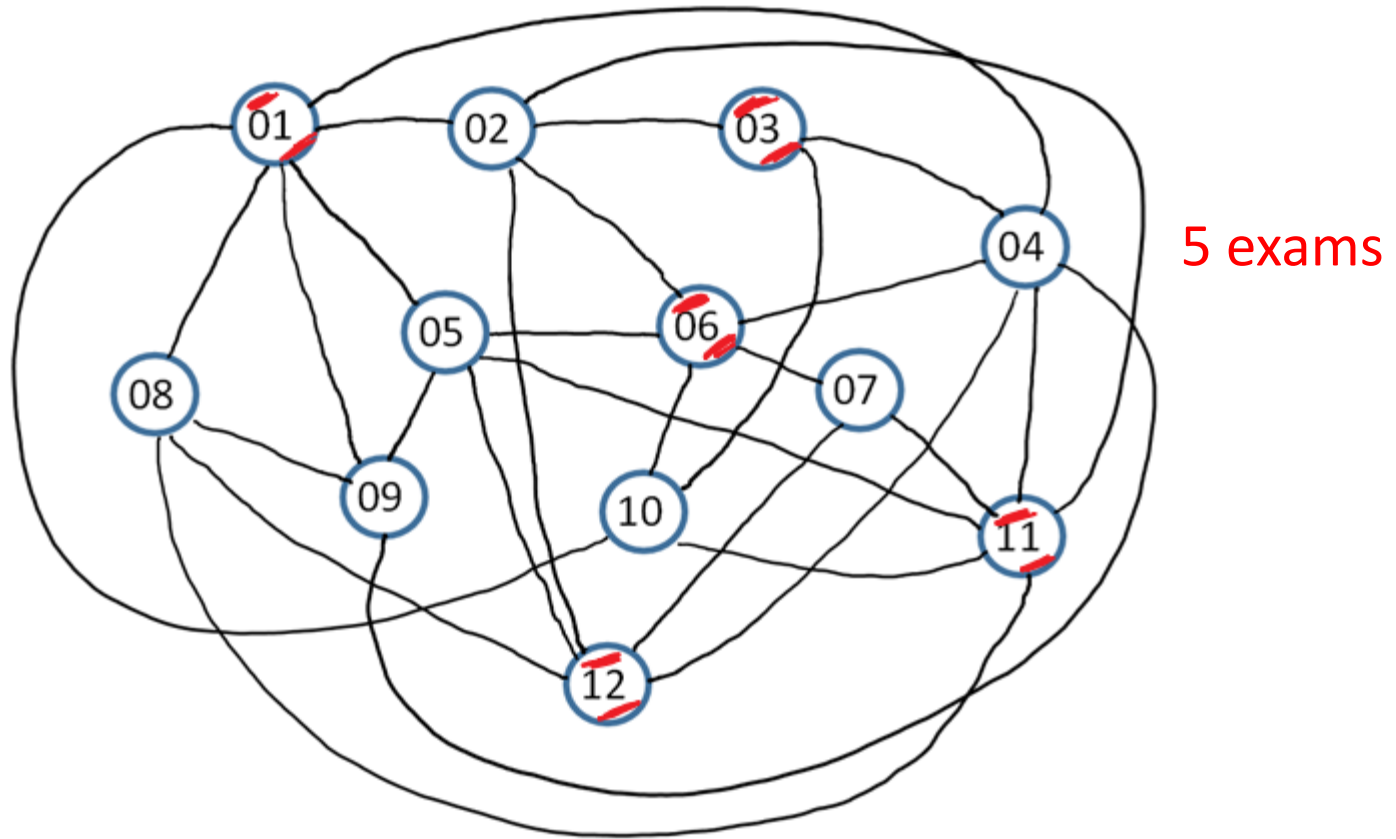


- Can you do better?
- How much better?
- You all get a copy of this graph on the handout.

You tell me, I draw...

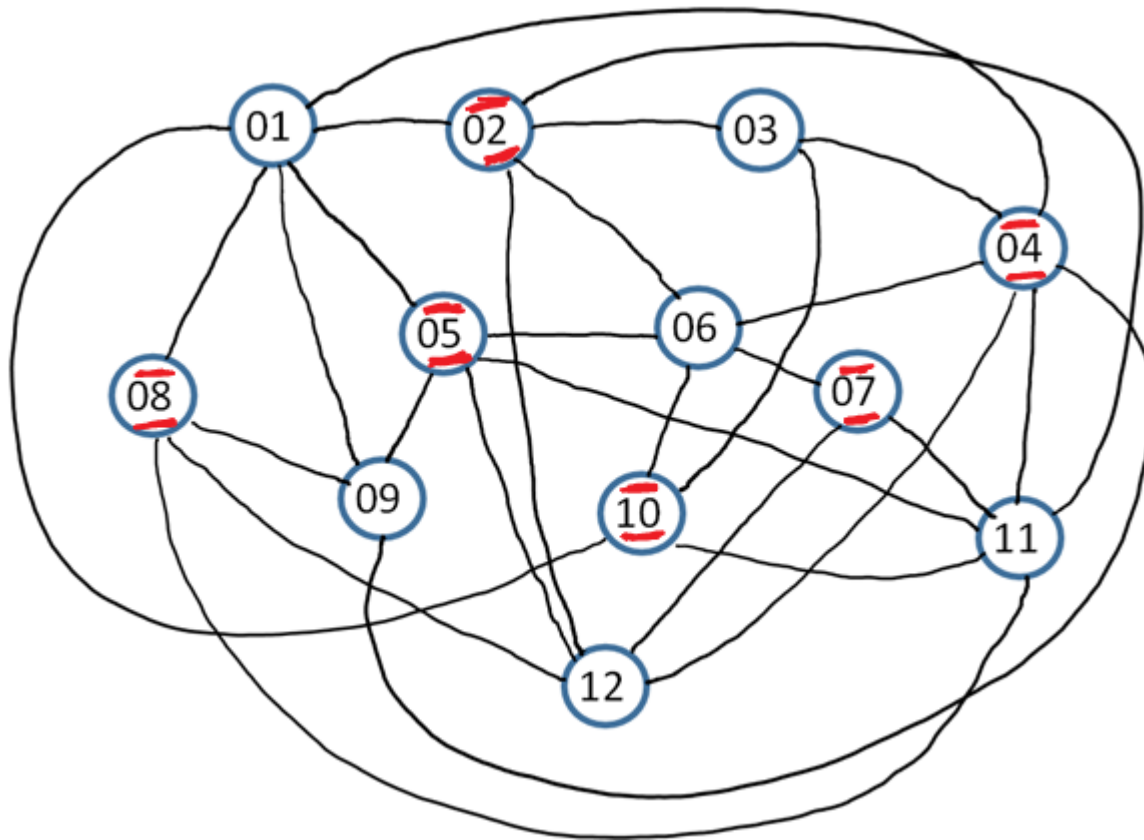


A couple of good attempts



■ Can you do better?

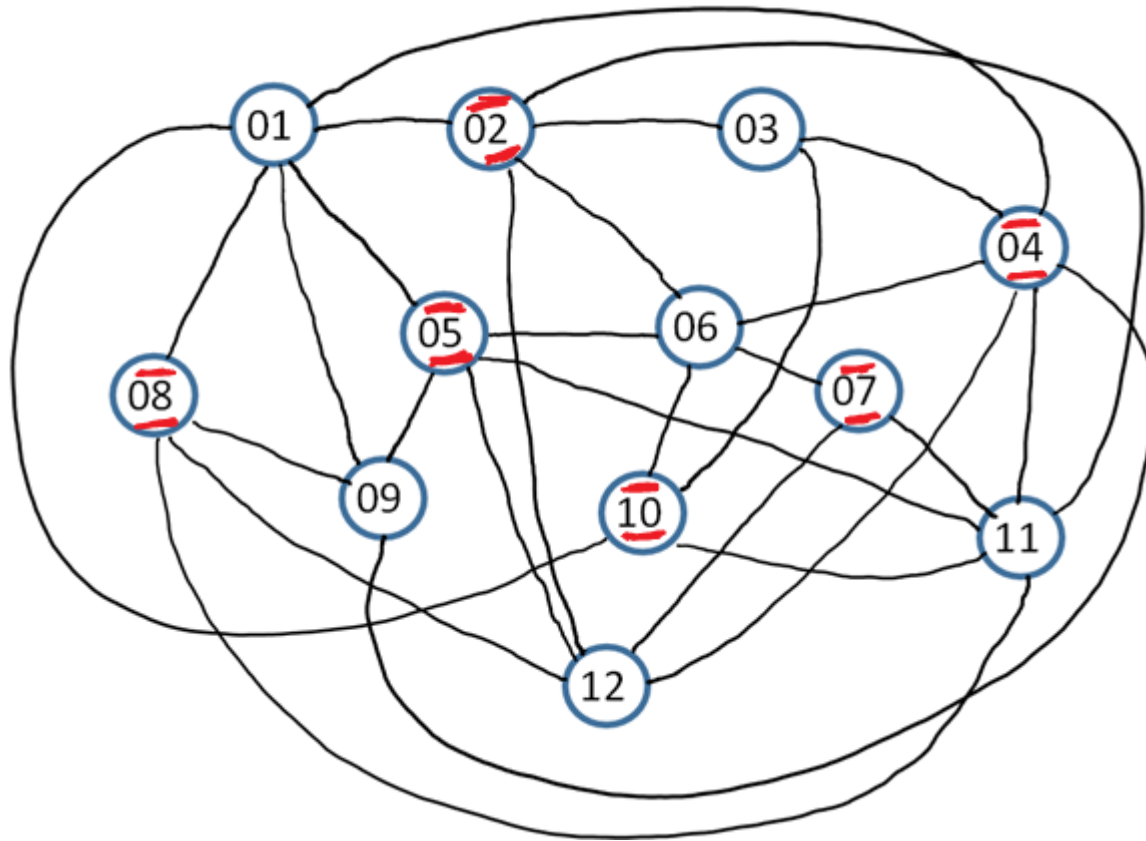
A couple of good attempts



6 exams

■ Can you do better?

A couple of good attempts



6 exams

- Tired of trying?
- Is this the best possible?

6 is the best possible!

- Proof by magic?



- Unfortunately not ☹

- No magician in the world has pulled out such a rabbit to this day! (By this we mean a rabbit that would work on *all* graphs.)

- Of course there is always a proof:

- Try all possible subsets of 7 nodes.

- There are 792 of them.

- Observe that none of them work.

- But this is no magic. It impresses nobody. We want a “short” proof. (We will formalize what this means.) Like the one in our max-flow example.

- Let’s appreciate this further...

Let's try another graph

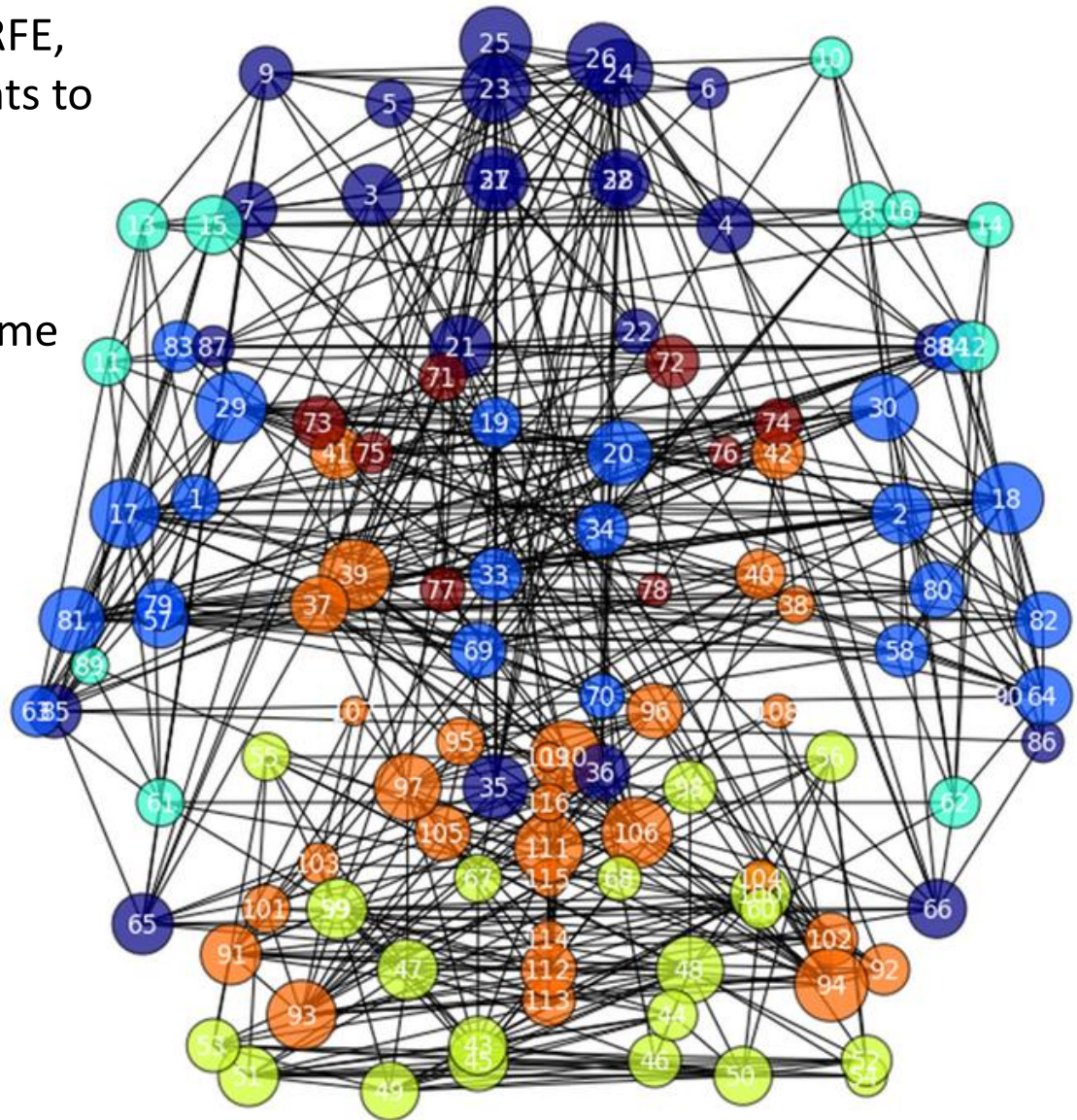
- Encouraged by the success of ORFE, now the Dean of Engineering wants to do the same for 115 SEAS courses.

- How many final exams on the same day are possible? Can you do 17?

 - You have 9 minutes! ;)

- Want to try out all possibilities for 17 exams?

- There are over 80000000000000000000 of them!



But there is some good news

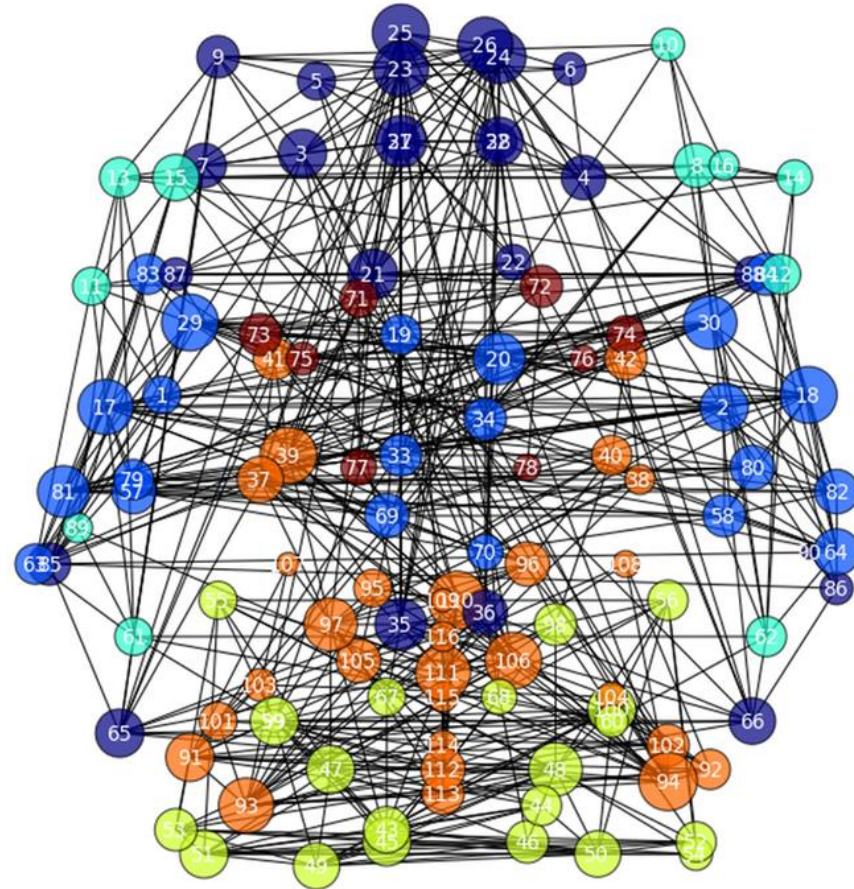
- Even though finding the best solution always may be too much to hope for, techniques from optimization (and in particular from the area of *convex optimization*) often allow us to find high-quality solutions with performance guarantees.

- For example, an optimization algorithm may quickly find 16 concurrent exams for you.

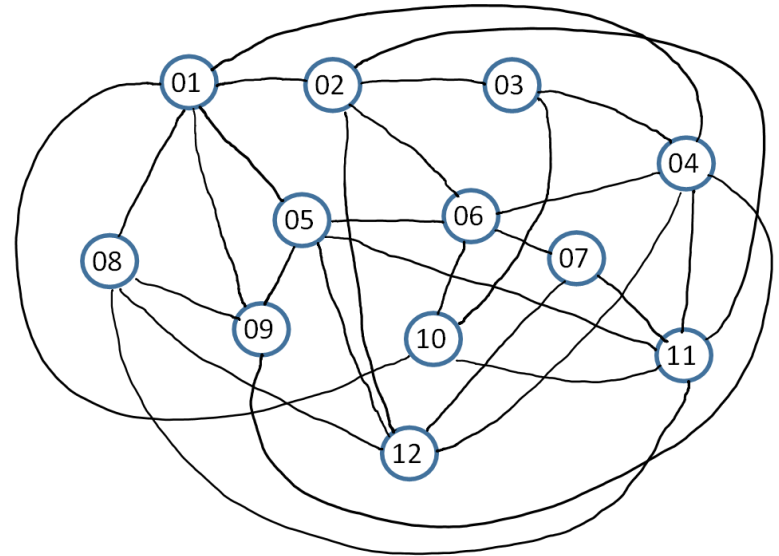
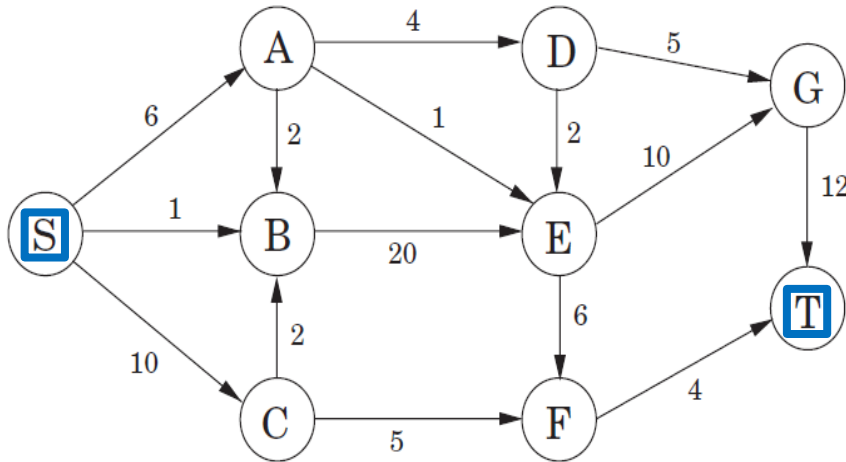
- You really want to know if 17 is impossible. Instead, another optimization algorithm (or sometimes the same one) tells you that 19 is impossible.

- This is very useful information! You know you got 16, and no one can do better than 19.

- We will see a lot of convex optimization in this class!



Which of the two problems was harder for you?

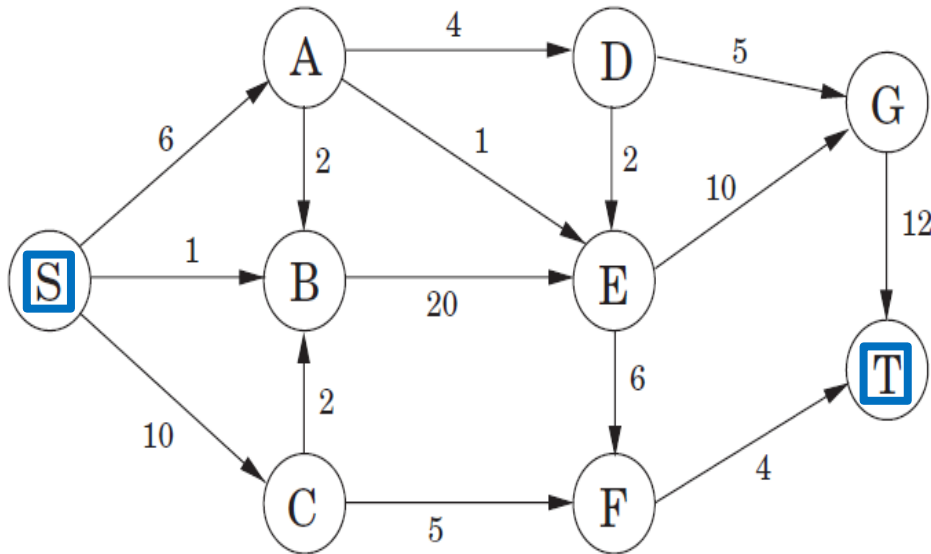


- Not always obvious. A lot of research in optimization and computer science goes into distinguishing the “tractable” problems from the “intractable” ones.
- The two brain teasers actually just gave you a taste of the **P vs. NP** problem. (If you have not heard about this, that’s OK. You will soon.)
- The first problem we can solve efficiently (in “polynomial time”).
- The second problem: no one knows. If you do, you literally get \$1M!
 - More importantly, your algorithm immediately translates to an efficient algorithm for thousands of other problems no one knows how to solve.



Modelling problems as a mathematical program

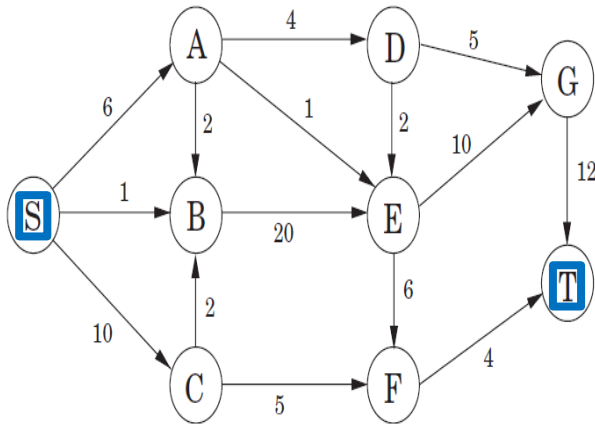
Let's revisit our first game



- What were your decision variables?
- What were your constraints?
- What was your objective function?

▪ Rules of the game:

- Cannot exceed capacity on the edges.
- For each node, except for S and T, flow in = flow out (i.e., no storage).
- **Goal:** ship as much oil as you can from S to T.



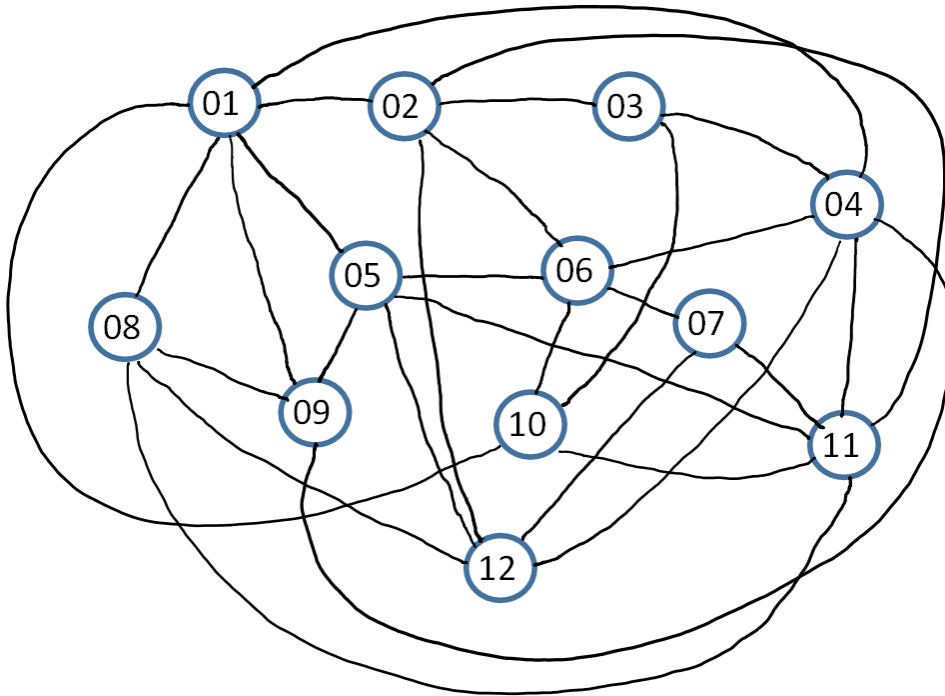
$x_{SA}, x_{AD}, x_{BE}, \dots, x_{GT}$ ← Decision variables

max. $x_{SA} + x_{SB} + x_{SC}$ ← Objective function

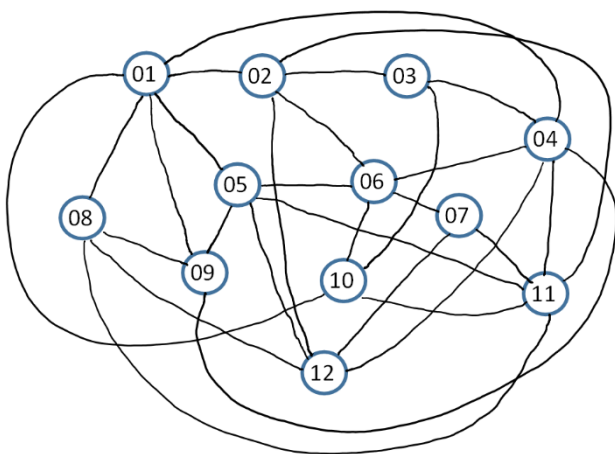
s.t.

- $x_{SA}, x_{AD}, x_{BE}, \dots, x_{GT} \geq 0$
 - $x_{SA} \leq 6, x_{AB} \leq 2, x_{EG} \leq 10, \dots, x_{GT} \leq 12$
 - $\begin{cases} x_{SA} = x_{AD} + x_{AB} + x_{AE} \\ x_{SC} = x_{CB} + x_{CF} \\ \vdots \\ x_{CF} + x_{EF} = x_{FT} \end{cases}$
- ← Constraints

Let's revisit our second game



- What were your decision variables?
- What were your constraints?
- What was your objective function?



x_1, x_2, \dots, x_{12} ← Decision variables

max. $x_1 + x_2 + \dots + x_{12}$ ← Objective function

s.t.

o $x_i (1 - x_i) = 0, i = 1, \dots, 12$

o $\left[\begin{array}{l} x_1 + x_2 \leq 1 \\ x_1 + x_8 \leq 1 \\ x_4 + x_6 \leq 1 \\ \vdots \\ x_{12} + x_8 \leq 1 \end{array} \right. \quad \text{(one per edge)}$

← Constraints

Why one hard and one easy? How can you tell?

$$x_{SA}, x_{AD}, x_{BE}, \dots, x_{GT}$$

$$\max. \quad x_{SA} + x_{SB} + x_{SC}$$

s.t.

$$\circ \quad x_{SA}, x_{AD}, x_{BE}, \dots, x_{GT} \geq 0$$

$$\circ \quad x_{SA} \leq 6, x_{AB} \leq 2, x_{EG} \leq 10, \dots, x_{GT} \leq 12$$

$$\circ \quad \begin{cases} x_{SA} = x_{AD} + x_{AB} + x_{AE} \\ x_{SC} = x_{CB} + x_{CF} \\ \vdots \\ x_{CF} + x_{EF} = x_{FT} \end{cases}$$

$$x_1, x_2, \dots, x_{12}$$

$$\max. \quad x_1 + x_2 + \dots + x_{12}$$

s.t.

$$\circ \quad x_i (1 - x_i) = 0, \quad i = 1, \dots, 12$$

$$\circ \quad \begin{cases} x_1 + x_2 \leq 1 \\ x_1 + x_8 \leq 1 \\ x_4 + x_6 \leq 1 \\ \vdots \\ x_{12} + x_8 \leq 1 \end{cases} \quad (\text{one per edge})$$

■ **Caution:** just because we can write something as a mathematical program, it doesn't mean we can solve it.

Fermat's Last Theorem

- Can you give me three positive integers x, y, z such that

$$x^2 + y^2 = z^2?$$

- Sure:

(3, 4, 5)	(5, 12, 13)	(8, 15, 17)	(7, 24, 25)
(20, 21, 29)	(12, 35, 37)	(9, 40, 41)	(28, 45, 53)

And there are infinitely many more...

- How about $x^3 + y^3 = z^3?$

- How about $x^4 + y^4 = z^4?$

- How about $x^5 + y^5 = z^5?$

Fermat's Last Theorem

- Fermat's conjecture (1637):

For $n \geq 3$, the equation $x^n + y^n = z^n$ has no solution over positive integers.

- Proved in 1994 (357 years later!) by Andrew Wiles.

(Was on the faculty in our math department until a few years ago.)



Arithmeticon Liber II.

61

interuallum numerorum 2. minor autem
1 N. atque ideo maior 1 N. + 2. Oportet
itaque 4 N. + 4. triplos esse ad 2. & ad-
huc superaddere 10. Ter igitur 2. adscit-
tis vnitatibus 10. æquatur 4 N. + 4. &
fit 1 N. 3. Erit ergo minor 3. maior 5. &
satisfaciunt quaestioni.

εἰ ἰδὼς ὁ ἀρετὴν μίσειεν ἴσται· εἰ ἰδὼς μὲ β. δὴ-
σει ἀρετὴν ἀειβυὸς δ' ἡμάρτανος δ' ἀντιπαραστήσας
τῷ μ. β. εἴ τι ὑπερίσχει μ. β. τρις ἀρε-
τῆς μάρτανος β. μὲ μ. β. ἴσται σὺν τῇ εἰς δ' ἡμάρτανον
δ. κ. γήνηται ἀειβυὸς μ. β. ἴσται ὡς ἡλῶ-
σαι μ. β. ὁ δὲ μίσειεν μ. β. κ. ποιῶσι τὸ
πορῶνται.

IN QVAESTIONEM VII.

CONDITIONIS appositæ eadem ratio est quæ & appositæ præcedenti quæstioni, nil enim aliud requirit quàm ut quadratus interualli numerorum sit minor interuallo quadratorum, & Canones iidem hic etiam locum habebunt, ut manifestum est.

QVÆSTIO VIII.

PROPOSITUM quadratum diuidere
in duos quadratos. Imperatum fit vt
16. diuidatur in duos quadratos. Ponatur
primus 1 Q. Oportet igitur 16 — 1 Q. qua-
le est quadrato. Fingo quadratum a nume-
ris quotquot libuerit, cum defectu con-
uinatur quod continet latus ipsius 16.
ello a 16. — 4. ipse igitur quadratus erit
4 Q. — 16. — 16 N. haec aequabunt vt
vitatibus 16 — 1 Q. Communis adiciatur
vtinque defectus, & a similibus auferen-
tur similia, sicut 5 Q. aequalis 16 N. & si-
1 N. 4. Erig igitur alter quadratorum
alter vtriusq; & vtinque summa est 5; seu
16. & vterque quadratus est.

ΤΟΝ ὑπερβόλον τετραμένον διελόν εἰς
 δύο τετραμένους. ἐπιτετραμένον δὲ τὸ
 διελόν εἰς δύο τετραμένους. καὶ τετραμένον
 σφραγισμένον διαμεινόμενον. διόντι δὲ μὴ
 δεῖς τὸ λήγειν διανάμεινον μὴ ἴσως τὴν
 τετραμένην. πλάσσειν τετραμένον εἰς δύο. δεῖς
 δὲ πότε λαίψει πούποτε μὴ δεῖς ὅτι μὴ ἴσως
 μὴ πλάσσειν. ἴσως εἴ βῃ λαίψει μὴ δὲ, αὐτὸς
 δεῖς δὲ πῶς αὐτὸν ἴσως διανάμεινον μὴ ἴσως
 λαίψει εἴ μὴ. πῶς τὴν μῆσιν λαίψει
 διανάμεινον μὴ. κοινὴ σφραγισμένη ἡ λαίψει,
 καὶ δὴ οὐκ ἔστιν ἔκστασις. διαμεινόμενος δὲ τὸ μὴ
 αὐλόμενος, καὶ ἡ γέννησις αὐτοῦ καὶ ἡ κηρυξίς.
 Τὸν ἴσως οὐκ ἐκκοσμήσινται. δὲ δὲ μὴ
 ἐκκοσμήσινται. Ἐοὶ δὲ δύο συνωπύηται πῶς
 ἐπὶ πῶς πῶς αὐτὸν.

OBSERVATIO DOMINI PETRI DE FERMAT.

Cubum autem in duos cubos, aut quadratoquadratum in duos quadratoquadratos
& generatim nullam in infinitum ultra quadratum potestatem in duos eius-
dem nominis fas est dividere cuius rei demonstrationem mirabilem sane detexi.
Hanc marginis exigentia non caperet.

QVÆSTIO IX.

R VRSVS oporteat quadratum 16
diuidere in duos quadratos. Ponatur
rursus primi lateris 1 N, alterius verò
quotocunque numerorum cum defectu tot
vnitatis, 2 quor. constar lateris diuidendi.
Eſto itaque 4 N. — 4. erunt quadrati, hic
quidem 1 Q. ille vero 4 Q. — 16. — 16 N.
Cæterum volo vtrumque ſimul æquari
vnitibus 16. Igitur 5 Q. — 16. — 16 N.
æquatur vnitibus 16. & ſit 1 N. 7 erit

ΕΣΤΩ δὴ πάλιν τὸν εἶς τετραγώνου διελ-
λῆναι εἰς δύο τετραγώνους. τετραγώνου πάλιν
ἢ τῷ πρώτῳ πλάτος εἰς ἑνὸς εἰς ἡ τῷ ὑπὲρ
εἰς ὅσον ἐκείνου λείψαι μ' ὅσον ἐστὶ ἡ τῷ δια-
γώνῳ πλάτος. Τῶν δὴ εἰς β' λείπει μ' δ'.
τοῦτον γ' τετραγώνου εἰς μὲν διαγώνῳ κῆτος
ὡς δὴ διευκλίδης δ' εἰς τ' λείπει εἰς η'. βῆ-
λουμεν τὸς δύο κῆτους συντείναντες. τῶν δὴ η'
εἰς. διευκλίδης ἀπὸ ε' μ' εἰς λείπει εἰς ι' καὶ
μ' η'. καὶ γίνονται οὗ ἀπὸ μὲν π. πεμπτέων.

H iii

Fermat's Last Theorem

■ Fermat's conjecture (1637):

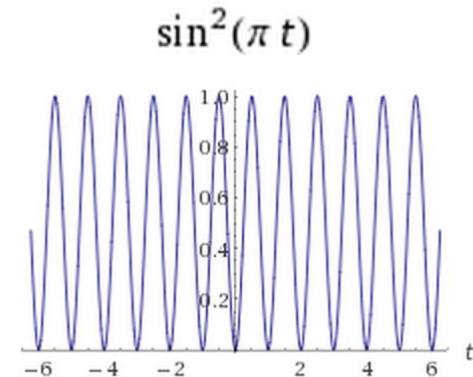
For $n \geq 3$, the equation $x^n + y^n = z^n$ has no solution over positive integers.

■ Consider the following optimization problem (mathematical program):

$$\min_{x, y, z, n} (x^n + y^n - z^n)^2$$

$$\text{s.t.} \quad x \geq 1, y \geq 1, z \geq 1, n \geq 3,$$

$$\sin^2 \pi n + \sin^2 \pi x + \sin^2 \pi y + \sin^2 \pi z = 0.$$



■ Innocent-looking optimization problem: 4 variables, 5 constraints.

■ If you could show the optimal value is non-zero, you would prove Fermat's conjecture!

Course objectives

- The skills I hope you acquire:

- Ability to view your own field through the lens of optimization and computation
 - To help you, we'll draw applications from operations research, statistics, finance, machine learning, engineering, ...
- Learn about several topics in scientific computing
- More mathematical maturity and ability for rigorous reasoning
 - There will be some proofs in lecture. Easier ones on homework.
- Enhance your coding abilities (nothing too fancy, simple MATLAB)
 - There will be a MATLAB component on every homework and on the take-home final.
- Ability to recognize hard and easy optimization problems
- Ability to use optimization software
 - Understand the algorithms behind the software for some easier subclass of problems.

Things you need to download

- Right away:

MATLAB

<http://www.princeton.edu/software/licenses/software/matlab/>

- In the next week or two (will appear on HW#2):

CVX

<http://cvxr.com/cvx/>

Course logistics

- On blackboard (and on blackboard).

- Image credits and references:

- [DPV08] S. Dasgupta, C. Papadimitriou, and U. Vazirani. Algorithms. McGraw Hill, 2008.
- [Sch05] A. Schrijver. On the history of combinatorial optimization (till 1960). In “Handbook of Discrete Optimization”, Elsevier, 2005.
<http://homepages.cwi.nl/~lex/files/histco.pdf>