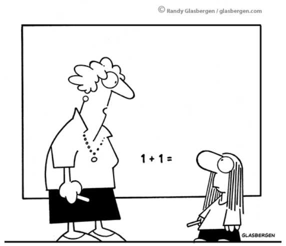
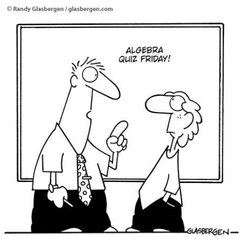
# Foundations of Computer Science
# Lecture 1

## Warmup: A Taste for Discrete Math and Computing

Background
Disease spread, speed-dating, friendship networks
3 Challenge Problems




"Yes, this will be useful to you later in life."




"It's important to learn math because someday you might accidentally buy a phone without a calculator."


"In 1953 you were my math teacher. You promised that algebra would come in handy someday. How much longer do I have to wait?"

## (Today) Warmup: A Taste for Discrete Math and Computing





## Resources and Rules

1. Web Page: www.cs.rpi.edu/~magdon/courses/focs.html
   – course info: www.cs.rpi.edu/~magdon/courses/focs/info.pdf
   – schedule+reading+slides: www.cs.rpi.edu/~magdon/courses/focs/slides.html
   – assignments+exams: www.cs.rpi.edu/~magdon/courses/focs/assign.html
2. Text Book: Discrete Mathematics and Computing (Magdon-Ismail).
3. TAs, UG-Mentors.
4. Recitation Section.
5. ALAC Drop-in-tutoring.
6. Professor.
7. **Prerequisites:**
   CS II (data structures)
   Calc I (Calc II STRONGLY recommended)
8. **Rules:** No food, no electronics, no cheating.



## The Storyline

1. Discrete objects.
2. Reasoning about discrete objects
3. Counting discrete objects
4. Randomness: probability
5. What can we compute?
6. What can we compute efficiently?

- concepts/concrete
- proof/theory/abstract
- theory of computation

our language will be mathematics . . .
. . . it will be everywhere

## Background

**Programming, numbers, geometry, algebra, calculus, . . .**

- What is the minimum element in the set $\{8, 9, 3, 10, 19\}$?
- Does this set of *positive* numbers have a minimum element:

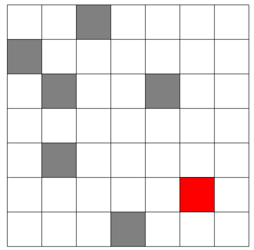
$$\{25, 97, 107, 100, 18, 33, 99, 27, 2014, 2200, 23, \ldots\}$$

**Any** (non-empty) set containing only **positive integers** has a minimum element.



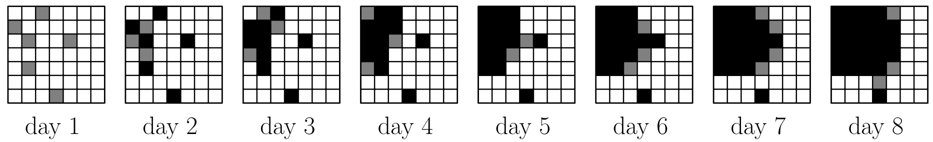
## Two-Contact Ebola on a Grid

A square gets infected if two or more neighbors (N,S,E,W) are infected.



- Given initial gray infections, who ultimately gets infected?
- Minimum infections to infect everyone?
- Given few vaccines, who to immunize?
- What were the "entry points"?

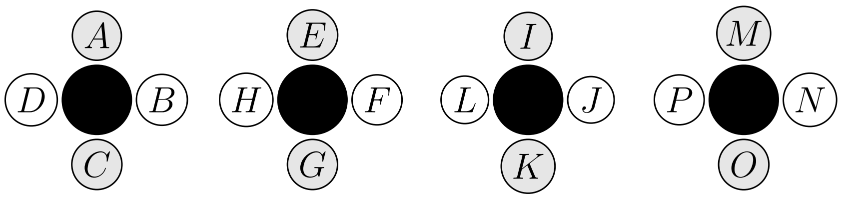
**Answers involve discrete math.**






## Scheduling Speed Dates

In each round 4 people "group"-speed-date around a table. (4 rounds in all)




How to organize the rounds so that people meet as many people as possible?

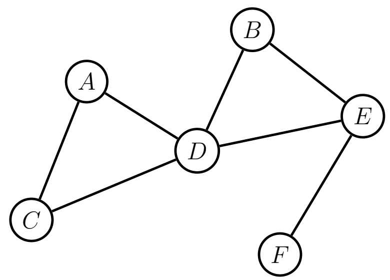
- Do you care about average or minimum number of meetups per person?
- Can everyone meet at least 10 people?
- What happens if you asign tables randomly?

**Answers involve discrete math.**



## Friendship Networks and Ads

People are circles and links are friendships.




Who would you advertise to? You wish to maximize adoption of your new technology.

**Answers involve discrete math.**

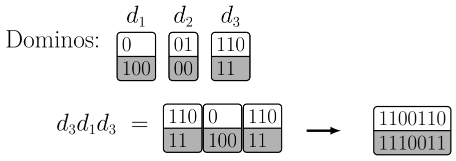
# Modeling Computers

Desktop, smartphone, fitbit, . . .

We have deep questions:

1. What can we compute?
2. What *can't* we compute?
3. How fast?

**What is computing?**




**Domino puzzle:** Want same top and bottom.

**Domino program:**
Input: dominos
Output: sequence that works
or
say it can't be done

**Answers involve discrete math.**



# Computing is Mathematics

"Too few people recognize that the high technology so celebrated today is essentially a mathematical technology."

"A programmer must *demonstrate* that their program has the required properties. If this comes as an afterthought, it is all but certain that they won't be able to meet this obligation. Only if this obligation influences the design is there hope to meet it. . .

"The required techniques of effective reasoning are pretty formal, but as long as programming is done by people who don't master them, the software crisis will remain with us and will be considered an incurable disease. And you know what incurable diseases do: they invite the quacks and charlatans in, who in this case take the form of Software Engineering Gurus."

– Edsger Dijkstra



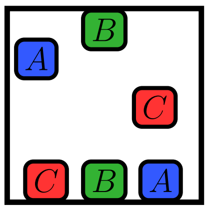
# Polya's Mouse

"A mouse tries to escape from an old fashioned cage. After many futile attempts bouncing back-and-forth, thumping his body against the cage bars, he finally finds one place where the bars are *slightly* wider apart. The mouse, bruised and battered escapes through this small opening, and to his elation, finds freedom." – Polya

Connect tiles of the same letter with wires. Wires cannot cross, enter tiles, or leave the box. How can it be done? If it can't be done, why not?

Don't be quick to dismiss either conclusion. Try this and that. Fiddle around until you understand the problem and the difficulty. Patience.

To solve such problems, "*You need brains and good luck. But, you must also sit tight and wait till you get a bright idea.*" – Polya.



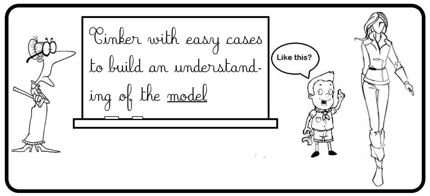


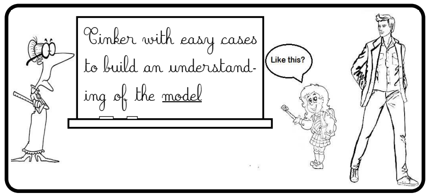

# Getting Good at Discrete Math

The *professional*'s workflow in addressing a discrete math problem:

1: Model the problem your are trying to solve using a discrete mathematical object.
2: Tinker with easy cases to build an understanding of the model.
3: Based on the tinkering, formulate a conjecture about your problem/model.
4: Prove the conjecture and make it a theorem. You now *know* something new.

**Tinker, Tinker, Tinker, Tinker!**



# Three Challenge Problems

## $100

*Distinct* subsets with the same sum

| | |
|---|---|
| 5719825393567961346558155629 | 1796439694824213266958886393 |
| 5487945882843158696672157984 | 6366252531759955676944496585 |
| 4767766531754254874224257763 | 8545458545636898974365938274 |
| 1855924359757732125866239784 | 3362291186211522318566852576 |
| 4289776424589197647513647977 | 8464473866375474967347772855 |
| 7967131961768854889594217186 | 2892857564355262219965984217 |
| 2572967277666133789225764888 | 4296693937661266715382241936 |
| 1294587141921952639693619381 | 8634764617265724716389775433 |
| 4764413635323911361699183586 | 8415234243182787534123894858 |
| 1474343641823476922667154474 | 2267353254454872616182242154 |
| 2578649763684913163429325833 | 4689911847578741473186337883 |
| 5161596985226568681977938754 | 4428766787964834371794565542 |
| 2242632698981685551523361879 | 7146295186764167268433238125 |
| 7474189614567412367516833398 | 2273823813572968577469388278 |
| 6211855673345949471748161445 | 6686132721336864457635223349 |
| 4942716233498772219251848674 | 3161518296576488158997146221 |
| 5516264359672753836539861178 | 1917611425739928285147758625 |
| 5854762719618549417768925747 | 3516431537343387135357237754 |
| 5313691171963952518124735471 | 7549684656732941456945632221 |
| 6737691754241231469753717635 | 2397876675349971994958579984 |
| 4292388614454146728246198812 | 4675844257857378792991889317 |
| 4468463715866746258976552344 | 2832515241382937498614676246 |
| 2638621731822362373162811879 | 8755442772953263299368382378 |
| 1258922263729296589785418839 | 9833662825734624455736638328 |
| 4482279727264797827654899397 | 5298671253425423454611152788 |
| 8749855322285371162986411895 | 9857512879181186421823417538 |
| 1116599457961971796683936952 | 1471226144331341144787865593 |
| 3879213273596322735993329751 | 3545439374321661651385735599 |
| 9212359131574159657168196759 | 6735367616915626462272211264 |
| 3351223183818712673691977472 | 2141665754145475249654938214 |
| 8855835322812512868896449976 | 8481747257332513758286947416 |
| 4332859486871255922555418653 | 9961217236253576952797397966 |
| 2428751582371964453381751663 | 9941237996445827218665222824 |
| 6738481866868951787884276161 | 6242177493463484861915865966 |
| 8794353172213177612939776215 | 4344843511782912875843632652 |
| 2989694245827479769152313629 | 7568842562748136518615117797 |
| 6117454427987751131467589412 | 2776621559882146125114473423 |
| 2761854485919763568442339436 | 6174299197447843873145457215 |
| 6884214746997985976433695787 | 5387584131525787615617563371 |
| 8671829218381757417536862814 | 5317693353372572284588242963 |
| 9431156837244768326468938597 | 6612142515552593663955966562 |
| 4788448664674885883585184169 | 1314928587713292493616625427 |
| 3624757247737414772711372622 | 2446827667287451685939173534 |
| 9361819764286243182121963365 | 9786693878731984534924558138 |
| 9893315516156422581529354454 | 2926718838742634774778713813 |
| 5913625989853975289562158982 | 3791426274497596641969142899 |
| 8313891548569672814692858479 | 2831727715176299968774951996 |
| 2265865138518379114874613969 | 3281287353463725292271916883 |
| 3477184288963424358211752214 | 9954744594922386766735519674 |
| 6321349612522496241515883378 | 3414339143545324298853248718 |

## $1,000

Domino Program

| $d_1$ | $d_2$ | $d_3$ |
|---|---|---|
| 0 | 01 | 110 |
| 100 | 00 | 11 |

$d_3d_1d_3$ = 

| | | |
|---|---|---|
| 110 | 0 | 110 |
| 11 | 100 | 11 |

→ 1100110 / 1110011

**Goal:** Want same top and bottom.

**Domino program:**

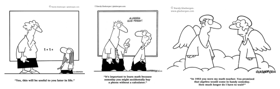
Input: dominos

Output: sequence that works
or
say it can't be done

## $10

Create the best 'math'-cartoon

Create a cartoon to illustrate some discrete math you learned in this class.



If you submit one, I can use it in the future