TABLE OF CONTENTS

PLUS SOLUTIONS TO ALL PROBLEMS

CHAPTER 1 COUNTING

SECTION 1.1 THE MULTIPLICATION RULE

the multiplication rule

Suppose you have 3 shirts (blue, red, green) and 2 pairs of pants (checked, striped). The problem is to count the total number of outfits.

The tree diagram (Fig 1) shows all possibilities: there are $2 \times 3 = 6$ outfits.

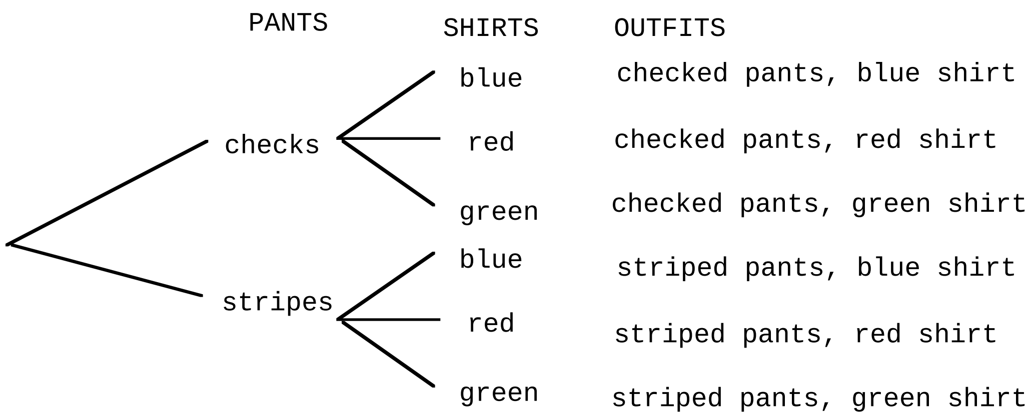


FIG 1

Instead of drawing the tree which takes a lot of space, think of filling a pants slot and a shirt slot (Fig 2). The pants slot can be filled in 2 ways, the shirt slot in 3 ways and the total number of outfits is the *product* 2×3.

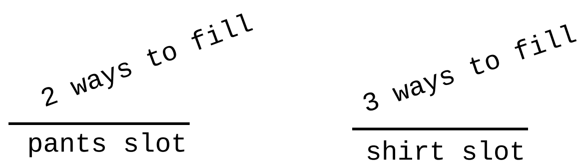


FIG 2 number of outfits = 3×2

(1) | If an event takes place in successive stages (slots), decide in how many ways each slot can be filled and then multiply to get the total number of outcomes |

example 1

You take a test with five questions. Each question can be answered True, False or left Blank. For instance some responses are

TTFBF
TFBBF
FTBBF

How many responses are there?

solution

There are five slots to fill (the five questions). Each can be filled in 3 ways (with T, F or B). So the number of responses is $3 \cdot 3 \cdot 3 \cdot 3 \cdot 3 = 3^5$

example 2

The total number of 4-letter words is $26 \cdot 26 \cdot 26 \cdot 26$ (each spot in the word can be filled in 26 ways).

example 3

The total number of 4-letter words that can be formed from 26 different scrabble chips is $26 \cdot 25 \cdot 24 \cdot 23$ (the first spot can be filled in 26 ways, the second in only 25 ways since you have only 25 chips left, etc.).

$7 \times 7 \times 7$ versus $7 \times 6 \times 5$

The answer $7 \times 7 \times 7$ is the number of ways of filling 3 slots from a pool of 7 objects where each object can be used over and over again; this is called sampling *with replacement*.

The answer $7 \times 6 \times 5$ is the number of ways of filling 3 slots from a pool of 7 objects where an object chosen for one slot cannot be used again for another slot; this is called sampling *without replacement*..

example 4

How many 4-letter words do not contain the letter Z.

solution

There are 4 slots and each can be filled in 25 ways. Answer is 25^4.

example 5

How many 4-letter words begin with TH.

solution

The first two letters are determined so there are only two slots to fill, the last two letters in the word. Each can be filled in 26 ways so the answer is $26 \cdot 26$.

counting the list "all, none, any combination"

Suppose you buy a car and are offered the options of Air conditioning, Power windows, Tinted glass, FM radio, Shoulder harnesses. Here are some of the possibilities:

1. none of the options
2. all of the options
3. just A
4. just P
5. just T
6. A and T
7. A, FM and T   etc.

The problem is to count the total number of possibilities.

Each of the five options is a slot which can be filled in two ways (yes or no). For example, buying A and T goes with

yes	no	yes	no	no
A slot	P slot	T slot	FM slot	S slot

The answer is $2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot = 2^5$.

(2) Given n objects, there are 2^n ways to choose all, none, any combination

example 6

There are six books $B_1, \ldots, B_6$ you are considering taking on your vacation. You can take them all; you can take none of them; you can take all except B_1 etc. How many possibilities are there.

solution

The number of possibilities is 2^6.

the number of subsets of a given set

If $Q = \{a,b,c\}$ then Q has 8 subsets:

$\{a\}$, $\{b\}$, $\{c\}$, $\{a,b\}$, $\{b,c\}$, $\{a,c\}$, $\{a,b,c\}$ and the null set ϕ

Look at the set $P = \{a_1,\ldots, a_6\}$.

To count the subsets think of 6 slots named $a_1,\ldots, a_6$, each to be filled with IN or OUT. For example the subset $\{a_2,a_3,a_5\}$ corresponds to

<u>OUT</u> <u>IN</u> <u>IN</u> <u>OUT</u> <u>IN</u> <u>OUT</u>

The null subset ϕ corresponds to

<u>OUT</u> <u>OUT</u> <u>OUT</u> <u>OUT</u> <u>OUT</u> <u>OUT</u> <u>OUT</u>

So there are 2^6 subsets of P. In general:

(3) $\boxed{\text{A set with } n \text{ members has } 2^n \text{ subsets}}$

choosing the best slots

Seven people $P_1,\ldots, P_7$ arrive in a city and they can stay at any of three hotels H_1, H_2, H_3. For instance, here are some of the possibilities.

1. all people in H_1
2. P_2 in H_1; others in H_2
3. P_1, P_5 in H_1; P_6 in H_2; others in H_3

I want to find the number of possibilities.

false start Try the hotels as slots, starting with H_1.
One possibility is that everyone stays at H_1.
Another possibility is that no one stays at H_1.
In fact any combination of the seven people could stay at H_1.
So by (2), the H_1 slot can be filled in 2^7 ways.
But then the H_2 slot depends on how the H_1 slot was filled.
If everyone stays at H_1 then there is only one possibility for the H_2 slot (no one stays there).
If no one stays at H_1 then by (2) again, there are 2^7 possibilities for H_2.
If $P_1,\ldots, P_6$ stay at H_1 then there are two possibilities for H_2 (P_7 or no one).
So there is no way to fill in a single value for the second slot.

good start Use the people as slots (since each person must get a hotel but not vice versa).
Each person-slot can be filled in 3 ways (with one of the 3 hotels).
The answer is 3^7.

example 7

An agency has 10 available foster families $F_1, \ldots, F_{10}$ and 6 children $C_1, \ldots, C_6$ to place. In how many ways can they do this if

(a) no family can get more than one child
(b) a family can get more than one child

solution

(a) Use the children as slots (since each child must get a family but not every family must get a child). Answer is $10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5$.

(b) Use the children as slots. Answer is 10^6.

example 8

Look at all 6-digit numbers which do not repeat a digit (e.g., 897563, 345678) Don't count 045678 which is really only a 5-digit number; i.e., leading 0's are not allowed.

How many are there?

solution

Fill six slots from the 10 available digits. But don't let the first slot be 0. Answer is $9 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5$.

using total – opposite

Suppose you want to count 4-letter words which contain repeated letters (e.g., ABZA, AACA, ZOGO, EEEE). It's easier to count the opposite (complementary) event, that is to count words *without* repeats:

words *with* repeats = total words − words with*out* repeats
$= 26^4 - 26 \cdot 25 \cdot 24 \cdot 23$

PROBLEMS FOR SECTION 1.1 (see the back of the notes for the solutions)

1. The library has 4 golf books, 3 tennis books and 6 swimming books. In how many ways can you bring home three books, one in each sport.

2. How many 6 digit numbers begin and end with 8.

3. There are 50 states and 2 U.S. Senators for each state. How many committees can be formed consisting of one Senator from each of the 50 states.

4. Ten people share a house. They need a cook for each day of the coming week. In how many ways can this be done if

(a) a person can be assigned to cook on more than one day
(b) a person can't get stuck with the job for more than one day

5. In how many ways can 7 people ride a long toboggan if only John and Mary are skilled enough to steer (the one who steers is the one up front).

6. Seven people fill out application forms. The company will hire those who turn out to be qualified (e.g., they might end up hiring John and Mary and rejecting the rest). How many possibilities are there.

7. In how many ways can the symbols A,B,C,D,E,E,E,E,E be arranged in a line so that no E is next to another E.

8. Four students are going to enroll in college and there are 11 school available. One possibility is that John, Mary, and Bill go to S_5 while Betty goes to S_{11}.

(a) How many possibilities are there.
(b) How many possibilities are there if the four students must go to four different schools.

9. In how many ways can the letters A,B,C, d,e,f,g be lined up if the capitals must come first.

10. (a) Three hundred people $P_1,\ldots, P_{300}$ make plane reservations but don't necessarily show up.

One possibility is the P_1 and P_2 show but the others are no-shows.

Another possibility is that P_1 and P_{300} show but the others are no-shows.

How many possibilities are there.

(b) Suppose the airline doesn't care about the *names* of the people who show up but just how *many* do. In that case the two possibilities listed in part (a) don't count as distinct since they both amount to 2 shows and 298 no-shows.

From this point of view how many possibilities are there.

11. Given ten boxes $B_1,\ldots, B_{10}$ and 17 balls $b_1,\ldots, b_{17}$. Toss the balls into the boxes.

One possibility for instance is

$$B_1 \text{ gets } b_1\text{–}b_{16}\ ,\ B_2 \text{ gets } b_{17} \quad \text{(other boxes are empty)}$$

How many possibilities are there.

12. (a) A hotel has 7 vacant single rooms $R_1,\ldots, R_7$. If five travelers $T_1,\ldots, T_5$ arrive, in how many ways can the desk clerk assign rooms.

(b) Repeat part (a) but with nine travelers (so that 2 will be turned away).

13. How many 4-letter words

(a) begin with Z

(b) begin and end with Z

(c) begin with Z and then contain no more Z's

14. The problem is to count six-digit even numbers with no repeated digits (e.g., 123456, 948712).

The number can't have a leading zero (can't be 045678) since then it really is only a 5-digit number, and it must end with an even digit to be an even number.

(a) Look at this attempt.

The first digit can be picked in 9 ways (any of the 10 digits excluding 0).
The next 4 digits can be picked in $9\cdot 8\cdot 7\cdot 6$ ways.
So far so good but what's the problem in picking the last digit?

(b) Try again!

The last digit can be picked in 5 ways (there are 5 even digits).
The middle 4 digits can be picked in $9\cdot 8\cdot 7\cdot 6$ ways.
So far so good but what's the difficulty in picking the first digit.

(c) Try again by breaking the original problem into cases to avoid the difficulties that turned up in (a) and (b).

15. In a certain programming language a name may consist of a single letter or a letter followed by up to 6 symbols which may be letters or digits (e.g., X, Z, R2D2, XX1234). How many names are there.

SECTION 1.2 PERMUTATIONS AND COMBINATIONS

review of factorials

$n! = n(n-1)(n-2)\dots1$ if n is a positive integer (definition)
$0! = 1$ (definition)

For example

$$6! = 6\cdot5\cdot4\cdot3\cdot2\cdot1 = 720$$

$$\frac{10!}{7!} = \frac{10\cdot9\cdot8\cdot7\cdot6\cdot5\cdot4\cdot3\cdot2\cdot1}{7\cdot6\cdot5\cdot4\cdot3\cdot2\cdot1} = 10\cdot9\cdot8 = 720$$

$$5! \times 6 = 6!$$

a deck of cards

A deck of cards contains 13 *faces* (values), namely Ace, King, Queen, Jack, 10,9,...,2 and four *suits*, namely Spades, Diamonds, Hearts, Clubs.
The *pictures* are Ace, King, Queen, Jack.
A *poker* hand contains 5 cards.
A *bridge* hand contains 13 cards.

permutations (lineups)

Consider 5 objects $A_1, \dots, A_5$.

To count all possible lineups (permutations) such as $A_1A_5A_4A_3A_2$, $A_5A_3A_1A_2A_4$ etc., think of filling 5 slots, one for each position in the line, and note that once an object has been picked, it can't be picked again. The total number of lineups is $5\cdot4\cdot3\cdot2\cdot1$ or $5!$.

In general:

$$\boxed{\text{n objects can be permuted in n! ways}}$$

Suppose you want to find the number of permutations of size 5 chosen from the 7 items $A_1, \dots, A_7$, e.g., $A_1A_4A_2A_6A_3$, $A_1A_6A_7A_2A_3$.

There are 5 places to fill so the answer is $7\cdot6\cdot5\cdot4\cdot3$, or, in more compact notation, $7!/2!$.

example 1

Ten people can be lined up (permuted) in $10!$ ways.
A permutation of 4 out of the 10 can be chosen in $10\cdot9\cdot8\cdot7$ ways.

matchups of same–size groups

Given 4 adults $A_1, \dots, A_4$ and 4 children $C_1, \dots, C_4$. Here are some possible matchups.

1. A_1 & C_4, A_2 & C_2, A_3 & C_3, A_4 & C_1
2. A_1 & C_3, A_2 & C_1, A_3 & C_4, A_4 & C_2

To count the total number of matchups take either the adults or the children to be the slots. If the adults are the slots then A_1 can be filled with any of 4 children, A_2 with any of the 3 remaining children etc. Answer is $4\cdot3\cdot2\cdot1$.

$$\boxed{\text{The number of ways to match two groups of size n is n!}}$$

combinations (committees)

Order doesn't count in a committee; it does count in a permutation.

A_1, A_{17}, A_2, A_{12} is the same committee as A_{12}, A_1, A_{17}, A_2 but $A_1A_{17}A_2A_{12}$ and $A_{12}A_1A_{17}A_2$ are different permutations.

The symbols $\binom{n}{r}$, called a *binomial coefficient*, stands for the number of committees of size r that can be chosen from a population of size n, or equivalently, the number of combinations of n things taken r at a time. Its value is given by

$$\binom{n}{r} = \frac{n!}{r!\ (n-r)!} \tag{1}$$

For example the number of 4-person committees that can be formed from a group of 10 is

$$\binom{10}{4} = \frac{10!}{4!\ 6!} = \frac{10\cdot 9\cdot 8\cdot 7\cdot 6\cdot 5\cdot 4\cdot 3\cdot 2\cdot 1}{4\cdot 3\cdot 2\cdot 1 \quad 6\cdot 5\cdot 4\cdot 3\cdot 2\cdot 1} = \frac{10\cdot 9\cdot 8\cdot 7}{4\cdot 3\cdot 2\cdot 1} = 10\cdot 3\cdot 7 = 210$$

notation $\binom{n}{r}$ (pronounced n on r) is also written as C(n,r).

why (1) works

I'll do it for a committee of size 4 from a population of $P_1, \ldots, P_{17}$.

The number of *lineups* of size 4 from $P_1, \ldots, P_{17}$ is $17\cdot 16\cdot 15\cdot 14$. To get the *committee* answer, compare lineups with committees.

list of committees	*list of lineups*	
(a) P_1, P_7, P_8, P_9	(a_1) $P_1P_7P_8P_9$	there are 4! of these
	(a_2) $P_7P_1P_9P_8$	
	$\vdots$	
	(a_{24}) $P_9P_9P_1P_8$	
(b) P_3, P_4, P_{12}, P_6	(b_1) $P_3P_4P_{12}P_6$	there are 4! of these
	(b_2) $P_4P_{12}P_3P_6$	
	$\vdots$	
etc.	(b_{24}) $P_{12}P_3P_4P_6)$	

Each committee gives rise to 4! lineups so

number of committees × 4! = number of lineups

$$\text{number of committees} = \frac{\text{number of lineups}}{4!} = \frac{17\cdot 16\cdot 15\cdot 14}{4!} = \frac{17!}{4!\ 13!} \quad \text{QED}$$

example 2

A library has 100 books. In how many ways can you check out 3 of them.

solution

Each threesome is a committee of 3 books out of 100. The answer is

$$\binom{100}{3} = \frac{100!}{3!\ 97!} = \frac{100\cdot 99\cdot 98}{3\cdot 2\cdot 1} = 161700$$

some properties of $\binom{n}{r}$

(2) $\boxed{\binom{n}{r} = \binom{n}{n-r}}$

For example $\binom{17}{4} = \binom{17}{13}$.

This holds because picking a committee of size 4 automatically leaves a committee of 13 leftovers and vice versa so the list of committees of size 4 and the list of committees of size 13 are the same length.

In particular $\binom{17}{4}$ and $\binom{17}{13}$ are each $\frac{17!}{4!13!}$.

(3) $\boxed{\binom{n}{1} = \binom{n}{n-1} = n}$ There are n committees of size 1 from a population of n.

(4) $\boxed{\binom{n}{n} = 1}$ There is one committee of size n from a population of n.

(5) $\boxed{\binom{n}{0} = 1}$ $\binom{n}{0} = \frac{n!}{n!\ 0!} = 1$ because 0! is defined as 1.

example 3

A poker hand is a committee of 5 chosen from 52 so there are $\binom{52}{5}$ poker hands.

example 4

How many poker hands contain the queen of spades.

solution

Since the hand must contain the spade queen, the problem amounts to picking the rest of the hand, a committee of size 4 from the 51 cards left in the deck.

Answer is $\binom{51}{4}$.

example 5

How many poker hands do *not* contain the queen of hearts.

sollution

method 1 Choose 5 cards from the 51 non-queen-of-hearts. Answer is $\binom{51}{5}$.

method 2

number of hands without queen of hearts

= total number of hands − number with queen of hearts

$= \binom{52}{5} - \binom{51}{4}$ (from examples 3 and 4)

example 6

How many poker hands contain only hearts.

solution

Choose a committee of 5 hearts from the 13 hearts in the deck. Answer is $\binom{13}{5}$.

example 7

How many poker hands contain only hearts and include the king of hearts.

solution

Choose 4 more hearts from the remaining 12 to go with the king. Answer is $\binom{12}{4}$.

example 8

How many poker hands contain only pictures but don't include the queen of spades.

solution

Choose 5 cards from the 15 non-spade-queen pictures

Answer is $\binom{15}{5}$.

example 9

How many poker hands contain 4 spades and the 2 of hearts.

solution

Choose 4 spades from 13. Answer is $\binom{13}{4}$.

$9 \times 8 \times 7$ versus $\binom{9}{3}$

Both count the number of ways in which 3 things can be chosen from a pool of 9. But $9 \times 8 \times 7$ corresponds to choosing the 3 things to fill slots (such as president, vice–president,secretary) while $\binom{9}{3}$ corresponds to the case where the 3 things are not given names or distinguished from one another in any way (such as 3 co–chairs)

From a committee/lineup point of view, $9 \times 8 \times 7$ counts lineups of 3 from a pool of 9 while $\binom{9}{3}$ counts committees.

notation

The symbol P(9,3) is often used for the number of permutations of 3 out of 9. So

$$P(9,3) = 9 \cdot 8 \cdot 7 = \frac{9!}{6!}$$

The symbols C(9,3) and Binomial[9,3] are often used for the number of committees of 3 out of 9. So

$$C(9,3) = \binom{9}{3} = \frac{9!}{3!\ 6!}$$

example 10

Given 3 girls and 7 boys. How many ways can 3 marriages be arranged.

For example, one possibility is

$$G_1 \ \& \ B_2, \quad G_2 \ \& \ B_7, \quad G_3 \ \& \ B_4$$

solution

method 1 Take the girls as slots. Answer is $7 \cdot 6 \cdot 5$.

method 2 First pick 3 of the 7 boys. Then match the 3 chosen boys with the 3 girls. Answer is $\binom{7}{3} \cdot 3!$.

The two answers agree since $\binom{7}{3} \cdot 3! = \frac{7!}{3!\ 4!}\ 3! = \frac{7!}{4!} = 7 \cdot 6 \cdot 5$

PROBLEMS FOR SECTION 1.2

1. An exam has 20 questions and you are supposed to choose any 10.
   (a) In how many ways can you make your choice.
   (b) In how many ways can you choose if you have to answer the first four questions as part of the 10.

2. How many poker hands contain
   (a) only spades
   (b) only pictures
   (c) no hearts
   (d) the ace of spades and the ace of diamonds (other aces allowed too)
   (e) the ace of spades and the ace of diamonds and no other aces

3. Four men check their hats and the hats are later returned to the men at random. One possibility is

$$M_1 \text{ gets } H_2,\quad M_2 \text{ gets } H_3,\quad M_3 \text{ gets } H_1,\quad M_4 \text{ gets } H_4$$

(a) How many possibilities are there.
(b) How many possibilities end with some dissatisfaction, i.e., some mismatch.

4. From a population of 10 men and 7 women how many ways are there to
(a) pick a King and Queen
(b) pick a president and vice-president
(c) pick two delegates
(d) award the good fellowship medal and the achievement medal

5. Compute
(a) $\frac{7!}{5!}$ (b) $\binom{7}{5}$ (c) $\binom{8}{5}/\binom{5}{2}$ (d) $\binom{12345}{1}$ (e) $\binom{12345}{0}$ (f) $\binom{12345}{12344}$ (g) $\binom{12345}{12345}$

6. Show that $\dfrac{\binom{5}{3}}{\binom{10}{3}}$ is the same as $\frac{5}{10}\,\frac{4}{9}\,\frac{3}{8}$.

7. Simplify $\dfrac{\binom{n+m-1}{n-1}}{\binom{n+m}{n}}$.

8. In how many ways can an employment agency match up 10 families and 6 nannies (with 4 families left over).

9. Consider committees of size 6 from the population $A_1, \ldots, A_{10}$.
(a) How many contain both A_1 and A_7.
(b) How many do not contain both A_1 and A_7.

SECTION 1.3 PERMUTATIONS AND COMBINATIONS CONTINUED

example 1

Find the number of poker hands with 2 Jacks.

solution

Pick 2 Jacks of the 4 Jacks

Pick the 3 remaining cards from the 48 non–Jacks

Answer is $\binom{4}{2}\binom{48}{3}$.

example 2

Find the number of poker hands with 2 Jacks and one Ace.

solution

Pick 2 of the 4 Jacks

Pick 1 of the 4 Aces

Pick the remaining 2 cards from the 44 non–Ace–non–Jacks

Answer is $\binom{4}{2}\cdot 4\cdot\binom{44}{2}$.

example 3

Find the number of committees of size 10 with 5 Men, 3 Women, 2 Children that can be picked from a population of 20 Men, 25 Women, 30 Children.

solution Pick 5 from the 20 Men.

Pick 3 from the 25 Women.

Pick 2 from the 30 Children.

Answer is $\binom{20}{5}\binom{25}{3}\binom{30}{2}$.

example 4

How many 8 digit strings contain exactly three 2's (e.g., 02322366).

solution

Pick 3 positions in the string for the 2's.

Fill the remaining 5 positions from the other 9 digits.

Answer is $\binom{8}{3}\cdot 9^5$.

example 5

How many poker hands contain only one suit.

solution

Can pick the suit in 4 ways. Can pick the 5 cards from that suit in $\binom{13}{5}$ ways.

Answer is $4\binom{13}{5}$.

example 6

Look at strings of digits and letters of length 7.

How many contain 2 digits, 4 consonants and 1 vowel if

(a) repetition is not allowed (e.g., z2ctaf3 is OK but z2zzaf3 is no good)

(b) repetition is allowed (e.g., 22zzyya is OK)

solution

(a) *method 1* Pick 2 digits, 4 consonants, 1 vowel and then line them up

Answer is $\binom{10}{2}\binom{21}{4}\cdot 5\cdot 7!$.

method 2 (perhaps better because it carries over into the case where repetition is allowed)

Pick 2 positions in the string for the digits. Can be done in $\binom{7}{2}$ ways.

Pick 4 positions for the consonants, leaving one for the vowel. $\binom{5}{4}$ ways.

Then fill the spots. Digit spots can be filled in $10\cdot 9$ ways, consonant spots can be filled in $21\cdot 20\cdot 19\cdot 18$ ways, vowel spot can be filled in 5 ways,

Answer is $\binom{7}{2}\binom{5}{4}\cdot 10\cdot 9\cdot 21\cdot 20\cdot 19\cdot 18\cdot 5$.

(b) The first method from part (a) doesn't work here.

First of all you don't have a rule yet for counting how many ways you can pick a committee of digits and letters when a digit or letter can be picked more than once.

Secondly, if the picks were say 2,7,Z,Z,Z,C,E you don't have a rule yet for permuting them because of the three identical Z's. And even if you had such a rule you would need cases because the picks might include repetition (e.g., 2,7,Z,Z,Z,C,E) or might not (e.g., A,B,C,5,D,3,E).

Here's a method that *will* work.

Pick two positions for the digits, four for the consonants, one for the vowel. Then fill the spots.

Answer is $\binom{7}{2}\binom{5}{4}\cdot 10^2 \cdot 21^4 \cdot 5$.

permutations with some objects kept together

Given ten people $P_1, \ldots, P_{10}$. In how many ways can they can be lined up so that P_2, P_6, P_9 are together; e.g., one possibility is

$$P_1 P_3 P_5 P_4 \boxed{P_2 P_9 P_6} P_{10} P_8 P_7$$

Temporarily think of the trio as a single item so that you have 7 loners and one trio. These 8 things can be lined up in 8! ways. Then rearrange the 3 within the trio in 3! ways.

Answer is 8! 3!.

permutations with some objects kept apart

Suppose you want to line up 7 girls and 3 boys so that no two boys are together.

Start by lining up the 7 girls. Can be done in 7! ways (Fig 1).

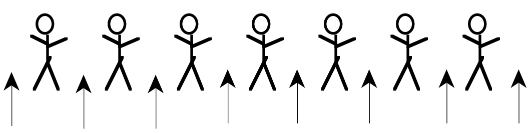


FIG 1

There are 8 spaces in between and at the ends in Fig 1. To keep the boys apart put the 3 boys in 3 of the 8 spaces. This can be done in $8\cdot 7\cdot 6$ ways (each boy is a slot) or alternatively in $\binom{8}{3}\cdot 3!$ ways (choose 3 spaces and then match them with the 3 boys).

Answer is $7!\cdot 8\cdot 7\cdot 6$ or equivalently $7!\binom{8}{3}\cdot 3!$.

To keep enemies apart, first line up the others and then select between/end places for the enemies.

example 7

How many permutations of the 26 letters of the alphabet have the vowels (A,E,I,O,U) together, have P,Q,R together and have X,Y,Z apart (i.e., no two of them together).

solution

First permute the 23 letters A–W with vowels together and P,Q,R together. To count these perms, line up a vowel clump, a PQR clump and the other 15 letters. That's a total of 17 things. Then permute within the vowel clump and within the PQR clump.

Can be done in 17! 5! 3! ways.

Now there are 18 between/ends where X,Y,Z could go. Pick places for them. Can be done in $18\cdot 17\cdot 16$ ways.

Final answer is $17!\ 5!\ 3!\ 18\cdot 17\cdot 16$

double counting

A poker hand with 2 pairs has say 2 Jacks, 2 Kings and then a fifth card which is not a Jack and not a King. I'll show you an incorrect way to count the number of poker hands with two pairs. The important thing it to understand *why* it's wrong.

(1) WRONG

step 1 Pick a face value and then pick 2 in that face.
step 2 Pick another face value and then pick 2 cards in that face.
step 3 Pick a 5th card from the 44 not of the two chosen faces (to avoid a full house).

Answer is $13 \cdot \binom{4}{2} \cdot 12 \cdot \binom{4}{2} \cdot 44$.

This is wrong because it counts the following as different outcomes when they are really the same hand:

outcome 1 Pick Queen, hearts and spades.
Pick Jack, hearts and clubs.
Pick Ace of clubs.

outcome 2 Pick Jack, hearts and clubs.
Pick Queen, hearts and spades.
Pick Ace of clubs.

This incorrect method uses "first face value" and "second face value" as slots but the faces can't be designated "first" and "second" so they can't be used as two slots.

The wrong answer in (1) counts every outcome twice.

Here's a correct version.
step 1 Pick a committee of 2 face values for the pairs.
step 2 Pick 2 cards from each face value.
step 3 Pick a fifth card from the 44 not of either face.
Answer is $\binom{13}{2}\binom{4}{2}\binom{4}{2} \cdot 44$.

Another correct method is to take the wrong answer from (1) and convert it to a right answer by taking half, i.e., the answer is $\frac{1}{2} \cdot 13 \cdot \binom{4}{2} \cdot 12 \cdot \binom{4}{2} \cdot 44$.

In general, an "answer" which *counts some outcomes more than once* is referred to as a *double count* (the double count in (1) happens to count *every* outcome *exactly* twice). Double counts can be hard to resist.

warning

A common mistake is to use say $7 \cdot 6$ instead of $\binom{7}{2}$, i.e., to implicitly fill slots named first thing and second thing when you should just be picking a committee of two things.

PROBLEMS FOR SECTION 1.3 (Make sure you look at the last problem.)

1. An exam offers 20 questions and you have to choose 2 of the first 5 and 8 from the next 15. In how many ways can you choose the 10 questions you'll answer.

2. In how many ways can you choose a committee of 2 men and 3 women from a group of 7 men and 8 women.

3. How many poker hands contain
 (a) 2 spades (meaning exactly 2 spades) one of which is the ace
 (b) 3 diamonds and 2 hearts
 (c) 4 black and 1 red
 (d) 2 aces (meaning exactly 2 aces)
 (e) 4 aces and no kings

4. How many license plates are there if a license plate contains
(a) 3 digits and 2 letters in any order(repetition allowed)
(b) 3 digits and 2 letters in any order, repetition not allowed
(c) 2 letters followed by 3 digits (repetition allowed)

5. Five people will be chosen from a group of 20 to take a trip.
 (a) In how many ways can it be done.
 (b) In how many ways can it be done if A and B refuse to travel together.
 (c) In how many ways can it be done if A and B refuse to travel without each other.

6. In how many ways can 20 books $B_1, \ldots, B_{20}$ be distributed among 4 children so that
 (a) each child gets 5 books
 (b) the two oldest get 7 books each and the two youngest get 3 each

7. Start with 3 men, M_1, M_2, M_3, 7 women, W_1, ..., W_7 and 8 children, C1, ..., C_8.
How many committees of size 5 can be formed containing
 (a) no men
 (b) M_3
 (c) M_3 and no other men
 (d) M_3 and two women
 (e) M_3, W_1, W_7
 (f) one man

8. Given 10 men, 20 women and 30 children.
(a) How many committees of size 6 contain 3 men.
(b) How many lineups of size 6 contain 3 men.

9. There are 50 states and 2 senators from each state. The President invites 25 senators to the White House. In how many ways can this be done so that 25 different states are represented.

10. In how many ways can you select 3 people from a group of 4 married couples C_1, C_2, C_3, C_4 so as to not include a pair of spouses.

11. (a) How many strings of 12 digits and/or letters contain 3 even digits
 (e.g., B57XYZZ$\boxed{4}\boxed{4}$C$\boxed{2}$3, 0429A^8).
 (b) Repeat part (a) if repetition is not allowed.

12. A child asks for a bicycle, a doll, a book and candy for her birthday. Assuming she'll get something but not necessarily everything she's asked for, how many possibilities are there.

13. A word is a palindrome if it reads the same backwards as forwards.
 (a) How many palindromes are there of length 5 (e.g., xyzyx, kkkkk)? length 6?
 (b) In a palindrome, some letters naturally must appear twice. If we insist that letters cannot appear *more* than twice how many palindromes are there of length 5? of length 6?

14. Start with 15 people. In how many ways can you
   (a) invite 4 for dinner
   (b) choose 4 to be president, vice-president, secretary and treasurer of your club
   (c) lend them 4 books titled A,B,C,D so that no person gets more than one book
   (d) lend them 4 books A,B,C,D allowing people to get more than one book
   (e) pick 5 to be on a basketball team and 4 to cheer for them.
   (f) pick 5 to play basketball in the winter and 4 to play football in the fall
   (g) divide them into 9 to play baseball, 2 to umpire and 4 to cheer

15. How many poker hands contain one ace.

16. How many permutations of A_1, A_2, A_3, A_4, B_1, B_2, B_3 have the A's in ascending order and the B's in ascending order (e.g., $A_1B_1B_2A_2A_3A_4B_3$, $A_1A_2B_1B_2A_3B_3A_4$).

17. There are 5 women $W_1, \ldots, W_5$ and 9 men $M_1, \ldots, M_9$ at a dance.

(a) In how many ways can the women choose dance partners (e.g., one possibility is W_1M_2, W_2M_7, W_3M_5, W_4M_4, W_5M_9 with wallflowers M_1, M_3, M_6, M_8).

(b) The answer to part (a) was not $5 \cdot 9$. What counting problem is $5 \cdot 9$ the answer to.

18. How many permutations of the letters A,B,C,D,E,F have B directly between A and C (e.g., D $\boxed{\text{CBA}}$ EF, $\boxed{\text{ABC}}$ FED).

19. You can order a hamburger with ketchup, mustard, pickle, relish, onion. How many possible hamburger orders are there.

20. How many poker hands contain
   (a) four aces
   (b) four of a kind (e.g., 4 Jacks)

21. A room has a row of 7 seats $S_1, \ldots, S_7$. Four people $P_1, \ldots, P_4$ sit down.

   (a) In how many ways can they be seated.
   (b) In how many ways can they be seated consecutively (i.e., with no empty seats in between).

22. How many committees of size 5 including a designated chairperson can be formed from a population of size 15.

23. 10 boys and 6 girls will sit in a row.
   (a) In how many ways can it be done if the 6 girls sit together.
   (b) In how many ways can it be done if no two girls sit together.
   (c) Why can't you do (b) by taking total − answer to (a).

24. Three Democrats, 5 Republicans, 6 Socialists and 7 uncommitteds will stage a (single-file) protest march. In how many ways can it be done if
   (a) each political group (but not the uncommitteds) sticks together
   (b) the Dems are first, Republicans next, Socialists next and uncommitteds last

25. How many permutations of the alphabet have
   (a) A,B,C together
   (b) D,E not together
   (c) A,B,C together and D,E together
   (d) A,B,C together and D,E not together

26. John and Mary play 5 sets of tennis. How many outcomes are there if
   (a) an outcome is a report of who wins what set (e.g., John wins 1st and 3rd and Mary wins the others)
   (b) an outcome is a report of who wins how many sets (e.g., Mary wins 3 sets and John wins 2 sets)

27. How many permutations of the 26 letters of the alphabet contain the string

mother (for example, the usual order abcde...xyz does not contain mother).

28. How many 3-card hands contain 3 of a kind (i.e., 3 of the same face) (e.g., Queen of hearts, spades, clubs; 3 of diamonds, clubs, spades).

29. (a) How many 4-card hands contain 2 pairs.
(b) How many 5-card hands contain a full house (3 of a kind and a pair).

30. A club with 50 members is going to form a finance committee.
(a) If the size hasn't been determined yet except for the fact that someone will be on it and not everyone will be on it, how many possibilities are there.
(b) How many possibilities are there if the committee will contain anywhere from 2 to 6 members.

31. Given 23 points in the plane.
(a) How many lines do they determine if no three of the points are collinear.
(b) What happens if the collinearity hypothesis is removed from part (a).

32. Look at samples of size 3 chosen from $A_1, \ldots, A_7$.
(a) How many samples are there if the samples are chosen *with replacement* (after an item is selected, it's replaced before the next draw so that A_1 for instance can be drawn more than once) and *order counts* (e.g., $A_2A_1A_1$ is different from $A_1A_2A_1$)
(b) How many samples are there if the drawing is *without replacement* and *order counts.*
(c) How many samples are there if the drawing is *w/o replacement* and *order doesn't count.*
(d) What's the 4th type of sample (counting here is tricky----coming up soon).

33. Here are some counting problems with proposed answers that double count. Explain *how* they double count (produce specific outcomes which are counted as if they are distinct but are really the same) and then get correct answers.
(a) To count the number of poker hands with 3 of a kind:
step 1 Pick a face value and 3 cards from that value.
step 2 Pick one of the remaining 48 cards not of that value (to avoid 4 of a kind).
step 3 Pick one of the remaining 44 not of the first or second value (to avoid 4 of a kind and a full house).
Answer is $13 \cdot \binom{4}{3} \cdot 48 \cdot 44$. WRONG

(b) To count the number of 3-letter words containing the letter z (e.g., zzz, zab, baz, zzc):
step 1 Pick one place in the word for the letter z (to be sure it appears).
step 2 Fill the other 2 places with any of the 26 letters (allowing more z's).
Answer is $3 \cdot 26 \cdot 26$. WRONG

(c) To count 7-letter words with 3A's:
step 1 Pick a spot for the first A.
step 2 Pick a spot for the second A.
step 3 Pick a spot for the third A.
step 4 Fill each of the remaining 4 places with any of the non-A's.
Answer is $7 \cdot 6 \cdot 5 \cdot 25^4$. WRONG

(d) To count 2-card hands not containing a pair:
step 1 Pick any first card.
step 2 Pick a second card from the 48 not of the first face value.
Answer is $52 \cdot 48$. WRONG

(e) To count the number of ways to form 3 coed couples from 10 men and 8 women:
step 1 Pick a man and woman for the first couple.
step 2 Pick a man and woman for the second couple.
step 3 Pick a man and woman for the third couple.
Answer is $10 \cdot 8 \cdot 9 \cdot 7 \cdot 8 \cdot 6$. WRONG

SECTION 1.4 PERMUTATIONS OF NOT-ALL-DISTINCT OBJECTS

permutations if not all objects are distinct (anagrams)

The number of permutations of

3 X's, 4 Y's, 6 Z's, and 1 each of A, B, C, D, E (18 letters total)

is $\dfrac{18!}{3!\ 4!\ 6!}$

> More generally, if among n objects there are
>
> n_1 identical ones of one type,
> n_2 identical ones of a second type
> $\vdots$
> n_5 identical ones of a fifth type
>
> and perhaps some other unique objects then the number of permutations (anagrams) of the n objects is
>
> (1) $$\frac{n!}{n_1!\ n_2!\ n_3!\ n_4!\ n_5!}$$

why (1) works

version 1

Look at permutations say of 3 A's, 2 B's, 2 C's D, E, F (10 letters total).
There are 10 spots on the line to fill.

Pick 3 of the spots for the A's. Can be done in $\binom{10}{3}$ ways.

Pick 2 spots for the B's from the remaining 7 places. Can be done in $\binom{7}{2}$ ways.

Pick 2 spots for the C's from the remaining 5 places. Can be done in $\binom{5}{2}$ ways.

The other 3 spots on the line and the 3 distinct letters can be matched in 3! ways.

$$\text{Number of perms} = \binom{10}{3}\binom{7}{2}\binom{5}{2}\,3! = \frac{10!}{3!\ 7!}\,\frac{7!}{2!\ 5!}\,\frac{5!}{2!\ 3!}\,3! = \frac{10!}{3!\ 2!\ 2!}$$

version 2

There are 10! perms of the 10 *distinct* letters A_1, A_2, A_3, B_1, B_2, C_1, C_2 D, E, F.

To get the number of permutations of the original 10 *not-all-distinct* letters, compare the following two lists.

perms of non-distincts	*perms of distincts*	
(a) AAABBFDECC	(a1) $A_1A_2A_3B_1B_2FDEC_1C_2$	there are 3! 2! 2! of these
	(a2) $A_2A_1A_3B_1B_2FDEC_1C_2$	
	(a3) $A_2A_1A_3B_2B_1FDEC_1C_2$	
	$\vdots$	
(b) ABABCCADEF	(b1) $A_1B_1A_2B_2C_1C_2A_3DEF$	there are 3! 2! 2! of these
	(b2) $A_2B_1A_1B_2C_1C_2A_3DEF$	
	(b3) $A_3B_2A_1B_1C_2C_1A_2DEF$	
	$\vdots$	
etc.		

Each perm on the left gives rise to $3!2!2!$ perms on the right. So

Number of perms of non-distincts $\times\ 3!2!2!$ = number of perms of distincts.

Number of perms of non-distincts $= \dfrac{10!}{3!\ 2!\ 2!}$ QED

example 1

How many permutations are there of the word APPLETREE.

solution

The 9-letter word contains 2 P's, 3 E's, A,L,R,T. Answer is $\dfrac{9!}{2!\ 3!}$.

example 1 continued

In how many of those permutations are the 3 E's together.

solution

Imagine that you have 7 objects, namely 2 P's, A,L,R,T and a clump of E's.

By (1) they can be permuted in $\dfrac{7!}{2!}$ ways

example 1 continued again

In how many of those permutations are the letters A, L, R together.

solution

Imagine that you have 7 objects, namely 2 P's, 3 E's, T and an ALR clump.

By (1) they can be permuted in $\dfrac{7!}{2!\ 3!}$ ways. And then the ALR clump can be rearranged in $3!$ ways.

Answer is $\dfrac{7!}{2!\ 3!} \cdot 3! = \dfrac{7!}{2!}$.

example 2

In how many ways can the complete works of Shakespeare in 10 volumes and 6 copies of Tom Sawyer be arranged on a shelf.

solution

This is a permutation of 16 objects, 6 of which are identical. Answer is $\dfrac{16!}{6!}$.

PROBLEMS FOR SECTION 1.4

1. How many permutations can be made using 3 indistinguishable white balls, 7 indistinguishable black balls and 5 indistinguishable green balls.

2. How many permutations are there of (a) HOCKEY (b) FOOTBALL

3. Look at all permutations of SOCIOLOGICAL
   (a) How many have the vowels adjacent
   (b) How many have the vowels in alphabetical order (but not necessarily adjacent); e.g., S A I C I L O C O G L O
   (c) How many have the vowels adjacent *and* in alphabetical order.

4. Suppose you want to count all 4-letter words containing 2 pairs (e.g., ADAD, QQSS, DAAD)

(a) What's wrong with the following attempt.

step 1 Pick a letter for the first pair. Can be done in 26 ways.

step 2 Pick a letter for the second pair. Can be done in 25 ways.

step 3 Arrange the 4 letters (2 of one kind and 2 of another kind).

Can be done in $\frac{4!}{2!\ 2!}$.

Answer is $26 \cdot 25 \cdot \frac{4!}{2!\ 2!}$. WRONG

(b) What's the right answer.

5. How many permutations of 4 plus signs, 4 minuses and 3 crosses have a plus in the middle (e.g., $+ + - \times - + \times + - \times -$).

6. Consider strings of length 10 from the alphabet A,B,C.

(a) How many are there.

(b) How many have 3 A's, 4 B's and 3 C's.

(c) How many have 3 A's.

7. Consider permutations of APPLETREE.

(a) How many have no adjacent E's.

(b) How many have none of A, L, R adjacent.

(c) How many have A, L, T together and no consecutive E's.

SECTION 1.5 COMMITTEES WITH REPEATED MEMBERS (INDISTINGUISHABLE BALLS INTO DISTINGUISHABLE BOXES)

the stars and bars formula

Consider committees of size 3 chosen from $A_1, \ldots, A_7$ with *repetition allowed*.

For example some possibilities are

A_2, A_2, A_3; i.e., 2 A_2's and an A_3

A_1, A_5, A_7; i.e., one each of A_5, A_1, A_5, A_7

Equivalently, consider unordered samples of size 3 chosen with replacement from a population of size 7.

I'll show that there are $\binom{7+3-1}{3} = \binom{9}{3}$ such committees.

The key idea (Fig 1) is that choosing a committee is like tossing 3 indistinguishable balls into 7 distinguishable boxes. For example, the committee A_5, A_5, A_6 corresponds to 2 balls in box A_5 and 1 ball in box A_6.

Furthermore the ball-in-boxes problem can be thought of as lining up 3 stars and 6 bars (the inside walls of the boxes). By (1) in the preceding section this can be done in $\frac{9!}{3!\,6!} = \binom{9}{6}$ ways.

sample *ball-in-box version* *stars and bars version*

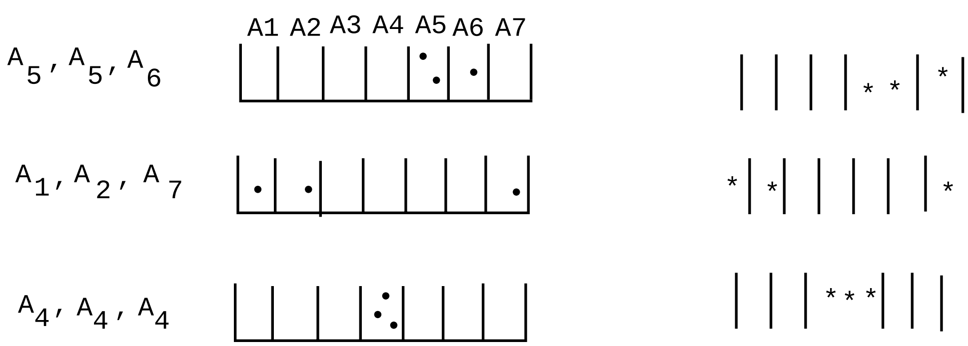


FIG 1

Here's the general result.

(1)	The following are the same: The number of committees of size k chosen from a population of size n with repetition allowed. The number of ways of tossing k indistinguishable balls into n distinguishable boxes. And that number is $\binom{n+k-1}{k}$

Note that since repetition is allowed on the committee, it's possible for k, the size of the committee, to be larger than n, the size of the population.

example 1

A school district encompasses four schools $S_1, \ldots, S_4$.

The district receives 9 blackboards.
In how many ways can the (identical) blackboards be distributed to the schools.

solution

The 4 schools are distinguishable boxes into which the 9 blackboards/balls are tossed. Answer is $\binom{4+9-1}{9} = \binom{12}{9}$.

warning

1. In example 1 the answer is *not* $\binom{4+9-1}{\boxed{4}}$. Of the two numbers 4 and 9, the one chosen for the lower spot in the formula is the one corresponding to the number of indistinguishable balls (i.e., to the size of the committee) and is not necessarily the smaller of the two numbers.
2. In example 1, the blackboards can't be used as slots since they are indistinguishable.

Indistinguishable objects can't serve as slots.

example 2

A child has a repertoire of 4 piano pieces $P_1, \ldots, P_4$.

Her parents insist that she practice 10 pieces a day. For example one practice session might consist of P_3 played 7 times (tra la la) and P_1 played 3 times.

(a) How many practice session are there.
(b) How many sessions are there if she must play P_2 at least 3 times.
(c) How many session are there if she must play P_2 exactly 3 times.
(d) How many sessions are there if she has to play each piece at least once.

solution

(a) Each session is a committee of size 10 chosen from the repertoire of size 4 with repetition allowed. Answer is $\binom{10+4-1}{10} = \binom{13}{10}$.

(b) After P_2 is played 3 times, the rest of the session is a committee of size 7 chosen from 4 (allow P_2 to be chosen again). Answer is $\binom{7+4-1}{7}$.

(c) After P_2 is played 3 times, the rest of the session is a committee of size 7 chosen from 3 (don't allow P_2 to be chosen again). Answer is $\binom{7+3-1}{7}$.

(d) Play each piece once. Then the rest of the session is a committee of 6 chosen from 4. Answer is $\binom{6+4-1}{6}$.

example 3

A bakery sells chocolate chip cookies, peanut butter cookies, sugar cookies and oatmeal cookies. In how many ways can you buy 7 cookies.

solution

You're choosing a committee of size 7 from a population of size 4 with repetition allowed. Answer is $\binom{7+4-1}{7}$.

example 4

A bakery has 27 chocolate chip cookies, 28 peanut butter cookies, 29 sugar cookies and 65 oatmeal cookies. In how many ways can you buy 7 cookies.

solution

This is the *same* as example 3. The numbers 27, 28, 29, 65 are irrelevant as long as they are ≥ 7 so that you can buy as many of each kind of cookie as you like.

See "at mosts" later in this section for what happens when they aren't irrelevant.

non-negative integer solutions to an equation of the form $x_1 + \ldots + x_n = k$

Look at the equation

$$x + y + z = 12$$

where x,y,z are non-negative integers. For example one solution is x=2, y=3, z=7; another solutions is x=0, y=0, z=12.

To count the number of solutions think of x,y,z as distinguishable boxes into which 12 indistinguishable balls are tossed. (The solution x=2, y=3, z=7 corresponds to 2 balls in the x box, 3 in the y box, 7 in the z box.) The answer is $\binom{3+12-1}{12}$.

at leasts

Here are three problems that are all the same:

> How many non-neg integer solutions to $x_1 + \ldots + x_4 = 18$ have $x_2 \geq 7$.
>
> If a bakery sells 4 kinds of cookies in how many ways can you choose 18 including at least 7 oatmeal
>
> In how many ways can 18 indistinguishable balls be tossed into 4 distinguishable boxes if the second box must get at least 7 balls.

For the first version, start x_2 at 7 and then count solutions to $x_1 + \ldots + x_4 = 11$.

For the second version buy 7 cookies and choose the remaining 11 from the 4 types of cookies

For the third version put 7 balls into box 2 and then toss the 11 others into the 4 boxes

The answer in each case is $\binom{11+4-1}{11}$.

at leasts continued

In how many ways can you toss 17 indistinguishable balls into 5 distinguishable boxes so that each box gets at least two balls.

Put two ball into each box (there's just one way to do this since the balls don't have names). Toss the remaining 7 balls into the 5 boxes.

Answer is $\binom{7+5-1}{7}$.

warning

This (easy) way of doing "at leasts" won't work with *distinguishable* balls or cookies. See Section 1.7 for what happens in that type of problem.

at mosts

A bakery sells chocolate cookies, peanut butter cookies, sugar cookies and oatmeal cookies.

You want to buy 18 cookies. In how many ways can it be done if the bakery has only 3 chocolates left (and an unlimited supply of the other 3 kinds).

This amounts to choosing a committee of size 18 from a population of size 4 with the restriction that the committee contain 3 or fewer chocolates, i.e., at most 3 chocs.

method 1

at most 3 chocs = 3 or fewer chocs = total − 4 or more chocs

The total is $\binom{18+4-1}{18}$.

To count "4 or more chocs" (i.e., at least 4 chocs), reserve 4 places on the committee for chocs and then pick 14 more from the 4 kinds (allowing more chocs); can be done in $\binom{14+4-1}{14}$ ways.

Answer is $\binom{18+4-1}{18} - \binom{14+4-1}{14}$.

method 2

N(at most 3 chocs) = N(no chocs) + N(1 choc) + N(2 chocs) + N(3 chocs)

$$= \binom{18+3-1}{18} + \binom{17+3-1}{17} + \binom{16+3-1}{16} + \binom{15+3-1}{15}$$

betweens

Toss 70 indistinguishable balls into 5 distinguishable boxes. In how many ways can it be done so that box 1 gets between 30 and 50 balls.

method 1

Between 30 and 50 in box 1 = 30 balls into box 1 + 31 balls into box 1
+ ... + 50 balls into box 1

For "30 balls into box 1", put 30 into box 1 and toss the other 40 into the remaining four boxes. Can be done in $\binom{40 + 4 - 1}{40}$ ways.

⋮

For "50 balls into box 1", put 50 into box 1 and toss the other 20 into the remaining four boxes. Can be done in $\binom{20 + 4 - 1}{20}$ ways.

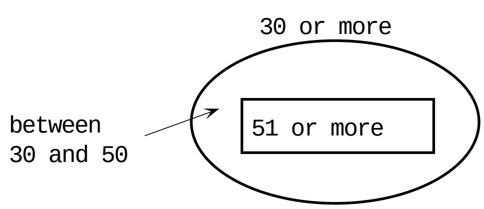
Answer is $\sum_{n=20}^{40} \binom{n + 4 - 1}{n}$

method 2 (better)

Look at the Venn diagram to see that

between 30 and 50 in box 1 = 30 or more in box 1 − 51 or more in box 1
= at least 30 in box 1 − at least 51 in box 1

For "30 or more", put 30 into box 1 and toss the other 40 into the *five* boxes.
Can be done in $\binom{40 + 5 - 1}{40}$ ways.
Similarly for "51 or more".
Answer is $\binom{40 + 5 - 1}{40} - \binom{19 + 5 - 1}{19}$



Check with Mathematica:

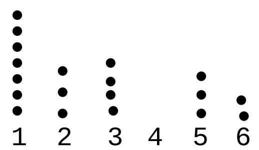
```
Binomial[40 + 5 - 1,40] - Binomial[19 + 5- 1,19]
126896

Sum[Binomial[n + 4 - 1,n],{n,20,40}]
126896
```

PROBLEMS FOR SECTION 1.5

1. A concert promoter has 1000 unreserved grandstand seats to distribute (free) among Alumni, Students, Faculty, Public. For example one possibility is to give half to the Students and half to the Faculty (Public and Alumni be damned). How many possibilities are there.

2. Toss a die 19 times. For each toss the die can come up 1,2,3,4,5 or 6. A histogram records how many times each outcome turns up. One possibility is shown in the diagram. How many possible histograms are there.



Problem 2

3. (a) In how many ways can 20 coins be picked from 4 large boxes filled respectively with pennies, nickels, dimes, and quarters.
(b) Why did part (a) say "large" boxes and how large is "large".

4. In how many ways can 12 balls be tossed into 5 distinct boxes if
(a) the balls are indistinguishable
(b) the balls are all different colors
(c) 8 of the balls are red (indistinguishable from one another) and 4 are white (indistinguishable from one another)

5. In how many ways can one quarter, one dime, one nickel and 25 pennies be distributed among 5 people.

6. A box contains balls B_1, B_2, ..., B_{10}. Draw 6. I don't care about the order in which they are drawn. How many possibilities are there if the drawing is
(a) without replacement (b) with replacement

7. You're taking six courses, $C_1, \ldots, C_6$.
For each course you will get a grade of A, B, C, D or E.
How many possible report cards can you get if
(a) the report lists the grade in each course so that one possibility is A's in C_1 and C_4, B in C_2, C's in the rest.
(b) the report is less detailed and only lists the overall results so that one possibility is 2 A's, 1 B, 3 C's

8. A car dealer has 5 models in stock: 7 M_1's, 8 M_2's, 9 M_3's, 10 M_4's and 12 M_5's.
Your company is going to buy 6 cars. For instance one possibility is 3 M_1's, an M_2 and 2 M_5's.
(a) How many possible ways are there to buy the 6 cars.
(b) How many possibilities include exactly 2 different models (e.g., 4 M_2's, 2 M_5's).
(c) How many possibilities include as many different models as possible.

9. You have 6 (identical) bats and 7 (identical) baseballs to give away to a group of 20 children.
(a) In how many ways can it be done.
(b) In how many ways can it be done if no child gets two balls and no child gets two bats.
(c) In how many ways can it be done if no child gets two gifts.

10. A message consists of all the 12 symbols $S_1, \ldots, S_{12}$ (in any order) plus 50 blanks distributed between the symbols (not before and not after, just between) so that there are at least 3 blanks between successive symbols. How many messages are there.

11. A child has a repertoire of 4 piano pieces and must practice 10 pieces at each session.

(a) How many different sessions are there.

(b) How many different sessions are there if she has to play each piece at least twice.

(c) Write out all the possibilities in (b) to confirm your answer.

(d) How many sessions are there if she has to play P_3 four times (exactly).

(e) How many sessions are there if she can play P_2 at most 3 times.

12. Toss 12 identical balls into boxes $B_1, \ldots, B_5$.

(a) In how many ways can it be done so that no box is empty.

(b) In how many ways can it be done so that B_2 gets an odd number of balls.

(c) Suppose you want to count the number of ways it can be done so that 3 boxes get all the balls; i.e., there are (exactly) 2 empties.

First explain what's wrong with this attempt.

Step 1 Pick 3 of the 5 boxes to get the balls

Step 2 Toss the 12 balls into those 3 boxes.

Answer is $\binom{5}{3}\binom{12+3-1}{12}$. WRONG

Then get a right answer.

13. A 5-sided die has sides named A, B, C, D, E. Toss it 100 times.

For instance some possible outcomes are

95 A's and 5 C's
1 A, 1 B, 98 D's
20 of each
etc.

How many of these outcomes have between 10 and 30 A's.
Do it in a way that doesn't involve a long sum.

14. A store sells 31 ice cream flavors.

In how many ways can you order a dozen cones if

(a) you want all different flavors

(b) exactly half must be chocolate

(c) you want exactly 7 different flavors

(d) the store has only enough strawberry for 2 cones

15. Consider non-negative integer solutions to $x_1 + x_2 + x_3 = 13$.

(a) How many are there.

(b) How many have $x_1 \ge 4$ and $x_2 \ge 2$.

(c) How many have $2 \le x_3 \le 5$.

(d) How many have $x_1 \ge 4$, $x_2 \ge 2$ and $2 \le x_3 \le 5$.

Suggestion. Take care of $x_1 \ge 4$, $x_2 \ge 2$, $x_3 \ge 2$ and then worry about $x_3 \le 5$.

SECTION 1.6 ORS

OR versus XOR

A OR B means A or B or both, called an *inclusive* or.

Similarly, A OR B OR C means one or more of A,B,C (i.e., exactly one of A,B,C or any two of A,B,C or all three of A,B,C).

On the other hand, A XOR B means A or B *but not both*, an *exclusive* or.
In these notes, *"or" will always mean the inclusive OR* unless specified otherwise.

In the real world you'll have to decide for yourself which kind is intended. If a lottery announces that any number containing a 6 or a 7 wins then you win with a 6 or 7 or both, i.e., 6 OR 7 wins (inclusive or). But if you order a coke or a 7-up you really mean a coke or a 7-up but not both , i.e., coke XOR 7-up.

OR rule (principle of inclusion and exclusion

Let N(A) stand for the number of ways in which event A can happen.

(1) $$\boxed{N(A \text{ or } B) = N(A) + N(B) - N(A \text{ and } B)}$$

(2) $$\begin{aligned} &N(A \text{ or } B \text{ or } C) \\ &\quad = N(A) + N(B) + N(C) - \big[\, N(A \,\&\, B) + N(A \,\&\, C) + N(B \,\&\, C) \,\big] + N(A \,\&\, B \,\&\, C) \end{aligned}$$

(3) $$\begin{aligned} &N(A \text{ or } B \text{ or } C \text{ or } D) \\ &\quad = N(A) + N(B) + N(C) + N(D) \\ &\qquad - \big[\, N(A \,\&\, B) + N(A \,\&\, C) + N(A \,\&\, D) + N(B \,\&\, C) + N(B \,\&\, D) + N(C \,\&\, D) \,\big] \\ &\qquad + \big[\, N(A \,\&\, B \,\&\, C) + N(A \,\&\, B \,\&\, D) + N(B \,\&\, C \,\&\, D) + N(A \,\&\, C \,\&\, D) \,\big] \\ &\qquad - N(A \,\&\, B \,\&\, C \,\&\, D) \end{aligned}$$

etc.

why (1) works

Suppose A can occur in 6 ways (the 4 x's and 2 y's in Fig 1). And suppose B can happen in 5 ways (the 2 y's and 3 z's in Fig 1). Then

$$N(A \text{ or } B) = 4 \text{ x's} + 2 \text{ y's} + 3 \text{ z's} = 9$$

On the other hand

$$N(A) + N(B) = 4 \text{ x's} + 2 \text{ y's} \quad + \quad 2 \text{ y's} + 3 \text{ z's} = 11$$

This is *not* the same as N(A or B) because it counts the y's *twice*. You *do* want to count them since this is an inclusive or, but we don't want to count them twice. So to get N(A or B), start with N(A) + N(B) and then subtract the number of outcomes in the intersection of A and B, i.e., subtract N(A & B) as in (1).

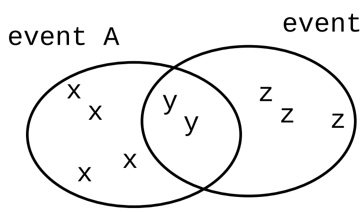


FIG 1

warning

The "or" in rule (1) is *inclusive*; it means A or B *or both*. You subtract away N(A & B) not because we want to throw away the boths but because you don't want to count them *twice*. In other words,

$$N(A \text{ or } B) = N(A \text{ or } B \textit{ or both}) = N(A) + N(B) - N(A \,\&\, B)$$

why (2) works

Suppose A can occur in 7 ways, B in 9 ways and C in 9 ways as shown in Fig 2.

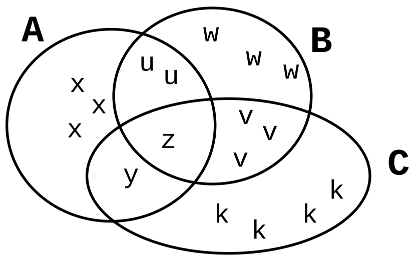


FIG 2

Then

$$N(A \text{ or } B \text{ or } C) = 3 \text{ x's} + 1 \text{ y} + 1 \text{ z} + 2 \text{ u's} + 3 \text{ w's} + 3 \text{ v's} + 4 \text{ k's} = 17$$

But N(A) + N(B) + N(C) isn't 17 because it counts the y, u's and v's twice each and it counts z three times.

Try again with

$$\underset{x,\,u,\,y,\,z}{N(A)} + \underset{u,\,w,\,v,\,z}{N(B)} + \underset{y,\,v,\,k,\,z}{N(C)} - \Big[\underset{u,\,z}{N(A \,\&\, B)} + \underset{y,\,z}{N(A \,\&\, C)} + \underset{v,\,z}{N(B \,\&\, C)}\Big]$$

Now the u's, v's, and y will be counted once each, not twice, but z isn't counted at all.

So formula (2) adds N(A & B & C) back in to include z again.

example 1

Find the number of bridge hands with 4 aces or 4 kings.

solution

By the OR rule,

$$N(4 \text{ aces or } 4 \text{ kings}) = N(4 \text{ aces}) + N(4 \text{ kings}) - N(4 \text{ aces and } 4 \text{ kings}).$$

When you count the number of ways of getting 4 aces don't think about kings at all (the hand may or may not include 4 kings ---- you don't care). The other 9 cards can be picked from the 48 non-aces so there are $\binom{48}{9}$ hand with 4 aces.

Similarly there are $\binom{48}{9}$ hands with 4 kings.

For 4K & 4A pick the other 5 cards from the 44 remaining cards.

Can be done in $\binom{44}{5}$ ways.

So $N(4 \text{ aces or } 4 \text{ kings}) = \binom{48}{9} + \binom{48}{9} - \binom{44}{5}$.

mutually exclusive (disjoint) events

Suppose A, B, C, D are mutually exclusive meaning that no two can happen simultaneously (i.e., A, B, C, D have no outcomes in common). Then all the "and" terms in (1), (2), (3) drop out and you get

(1') N(A or B) = N(A) + N(B)
(2') N(A or B or C) = N(A) + N(B) + N(C)
(3') $N(A_1$ or ... or $A_n) = N(A_1) + \ldots + N(A_n)$

example 2

Count poker hands containing all spades or all hearts.

solution

The events "all spades" and "all hearts" are mutually exclusive since they can't happen simultaneously. So by (1'),

N(all spades or all hearts) = N(all spades) + N(all hearts) = $\binom{13}{5} + \binom{13}{5}$

warning

N(A or B) is *not* N(A) + N(B) *unless* A and B are mutually exclusive. If they aren't, *don't forget to subtract N(A & B)*.

PROBLEMS FOR SECTION 1.6

1. How many poker hands have 2 aces or 3 kings.

2. How many poker hands have the ace of spades or the king of spades.

3. How many 7-digit strings have two 3's or two 6's.

4. Suppose you're computing $N(A_1$ or A_2 or ... or $A_8)$.
(a) Eventually you have to add in 2-at-a-time terms like $N(A_1 \& A_5)$, $N(A_3 \& A_5)$ etc. How many of these terms are there.

(b) Eventually you have to add in 3-at-a-time terms like $N(A_1 \& A_4 \& A_7)$, $N(A_3 \& A_7 \& A_8)$ etc. How many of these terms are there.

5. How many poker hands have (a) 2 aces or 2 kings (b) 3 aces or 3 kings

6. How many poker hands have no spades or no hearts.

7. Toss a die 7 times. Order doesn't count so that one outcome for instance is 1 two, 4 fives, 2 sixes.
(a) How many outcomes are there.
(b) How many outcomes have 1 one or 2 twos or 3 threes.

8. A group consists of 6 children, 7 teenagers, 8 adults. How many committees of size 3 contain only one age group.

9. How many poker hands contain the Jack of Spades XOR the Queen of Spades (i.e., Jack or Queen but *not* both).

SECTION 1.7 AT LEASTS

at least 4 chocolates in a paper bag (i.e., at leasts for committees with repeated members allowed) (i.e., at leasts for indistinguishable balls into distinguishable boxes) (Section 1.5)

A store sells 5 ice cream flavors. Here's how to find the number of ways to buy 12 cones so as to get at least one chocolate.

Include one chocolate and then choose the rest of the committee, 11 cones, from the 5 flavors (this allows more chocs). Answer is $\binom{11+5-1}{11}$.

WRONG WAY to do at leasts for ordinary committees (no repeated members)

The method that just worked with the choc cones (put one choc on the committee to be sure) does *not* work here.

Here's an example to show why not.

Suppose you want to count poker hands with at least one ace (all cards are distinguishable from one another). Try the "pick one to be sure" method.

WRONG

step 1 Pick one ace to be sure of getting at least one. Can be done in 4 ways.

step 2 Pick 4 more cards from the remaining 51. Can be done in $\binom{51}{4}$ ways.

(The 51 allows more aces. If you use 50 you are finding *exactly* one ace)

"Answer" is $4\binom{51}{4}$.

The answer is wrong because it counts the following as different outcomes when they are really the same.

outcome 1 *step 1* Pick the spade ace as the sure ace.
step 2 Pick the ace, king, queen, jack of clubs.

outcome 2 *step 1* Pick the club ace as the sure ace.
step 2 Pick the spade ace and the king, queen, jack of clubs.

So this attempt *double counts* (i.e., counts some outcomes more than once).

Some RIGHT WAYS to do at leasts for ordinary committees

Here are some correct methods for finding N(at least one ace).

method 1

$$N(\text{at least one ace}) = \text{total} - N(\text{no aces}) = \binom{52}{5} - \binom{48}{5}$$

method 2

$$N(\text{at least one ace}) = N(1A \text{ or } 2A \text{ or } 3A \text{ or } 4A)$$

The events 1A (meaning exactly one ace), 2A, 3A, 4A are mutually exclusive so you can use the abbreviated OR rule.

$$N(\text{at least one ace}) = 1A + 2A + 3A + 4A = 4\binom{48}{4} + \binom{4}{2}\binom{48}{3} + \binom{4}{3}\binom{48}{2} + 48$$

method 3

$$N(\text{at least one ace}) = N(A_S \text{ or } A_H \text{ or } A_C \text{ or } A_D)$$

$$= \Big[N(A_S) + N(A_H) + N(A_C) + N(A_D)\Big] - \Big[N(A_H \,\&\, A_C) + \text{other 2-at-a-time terms}\Big]$$

$$+ \Big[N(A_C \,\&\, A_H \,\&\, A_D) + \text{other 3-at-a-time terms}\Big] - N(A_S \,\&\, A_H \,\&\, A_C \,\&\, A_D)$$

The first bracket contains 4 terms all having the value $\binom{51}{4}$.

The second bracket contains $\binom{4}{2}$ terms each of which has value $\binom{50}{3}$.

The third bracket contains $\binom{4}{3}$ terms each of which has value $\binom{49}{2}$.

So N(at least one ace) = $4\binom{51}{4} - \binom{4}{2}\binom{50}{3} + \binom{4}{3}\binom{49}{2} - 48$

In this example, method 1 was best but you'll see examples favoring each of the other methods (cf. problem 5)

footnote

The "pick one to be sure" method that works for committees with repetition allowed does not work for ordinary committees. But methods 1 and 2 that worked for the ordinary committees will also work for committees with repetition.

example 1

In how many ways can 10 sandwiches be given to 4 people so that Mary gets at least 3 sandwiches if

(a) the sandwiches are all tuna (indistinguishable)

(b) the sandwiches are tuna, ham, jelly,... (all different from one another)

solution (a) This is tossing 10 indistinguishable balls into 4 distinguishable boxes (same as committees with repetition allowed).

Give Mary 3 tunas. Then toss the remaining 7 tunas into the 4 distinguishable people. Answer is $\binom{7+4-1}{7}$.

(b) total − N(Mary gets none or one or two)

$= \text{total} - \Big[\text{N(Mary gets none)} + \text{N(Mary gets 1)} + \text{N(Mary gets 2)}\Big]$

To find the total, think of each sandwich as a slot which can be filled in 4 ways.

To find N(Mary gets none), fill each sandwich slot in 3 ways.

For N(Mary gets 1), pick a sandwich for Mary (can be done in 10 ways) and then fill each of the remaining 9 sandwich slots in 3 ways (can be done in 3^9 ways).

For N(Mary gets 2), pick 2 sandwiches for Mary (can be done in $\binom{10}{2}$ ways and then fill each of the remaining sandwich slots in 3 ways (can be done in 3^8 ways).

Answer is $4^{10} - \Big[3^{10} + 10\cdot 3^9 + \binom{10}{2}\cdot 3^8\Big]$.

example 2

How many 7-digit strings have at least five 3's (e.g., 3376333, 3133333).

solution N(at least five 3's) = N(five 3's) + N(six 3's) + N(seven 3's)

To find N(five 3's) pick five places for the 3's and fill the other 2 places with non-3's.

To find N(six 3's) pick six places for the 3's and fill the one remaining place with a non-3. So

$$\text{N(at least five 3's)} = \binom{7}{5}9^2 + \binom{7}{6}9 + 1$$

wrong way to do example 2

If you pick 5 places in the string for 3's (to be sure) and then fill the other places with any of 10 digits you get "answer" $\binom{7}{5}\,10^2$ and it's WRONG.

It's wrong because it counts the following as different outcomes when they are really the same (they are both the string 3333339):

outcome 1 step 1 Pick positions 1−5 for the sure 3's
step 2 Fill position 6 with a 3 and position 7 with a 9.

outcome 1 step 1 Pick positions 1−4, 6 for the sure 3's
step 2 Fill position 5 with a 3 and position 7 with a 9

> Except for committees with repeated members (i.e., tossing indistinguishable balls into distinguishable boxes) you can't do "at least n of them" by starting with n "sure" ones and then picking the rest.

at mosts

Here's how to find the number of bridge hands with *at most* 10 spades.

N(at most 10 spades = total − N(11 spades or 12 spades or 13 spades)
= total − N(11 spades) − N(12 spades) − N(13 spades)

$$= \binom{52}{13} - \binom{13}{11}\binom{39}{2} - \binom{13}{12}39 - 1$$

exactlies combined with at leasts

I'll find the number of poker hands with (exactly) 2 spades and at least 1 heart.

method 1 Within the 2-spade world, some hands have at least one heart and the rest have no hearts.

The number of poker hands with 2 spades *and* at least 1 heart is region II in Fig 1. You can find it by taking the entire 2-Spade region and subtracting region I. So

N(2S and at least 1H) = N(2S) − N(2S and no H)

$$= \binom{13}{2}\binom{39}{3}\ \begin{matrix}\text{pick 2 spades}\\ \text{pick 3 others}\end{matrix} \quad - \quad \binom{13}{2}\binom{26}{3}\ \begin{matrix}\text{pick 2 spades}\\ \text{pick 3 non-hearts}\end{matrix}$$

warning N(2 spades at at least 1 H) is *not* N(2 spades) − N(no H).
It is N(2 spades − N(*2 spades* and no H)

2 Spades

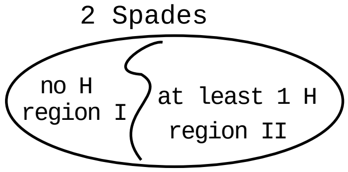


FIG 1

method 2

N(2S and at least 1H) = N(2S & 1H) + N(2S & 2H) + N(2S & 3H)

$$= \binom{13}{2}\cdot 13\cdot\binom{26}{2} + \binom{13}{2}\binom{13}{2}\cdot 26 + \binom{13}{2}\binom{13}{3}$$

at least one of each

Choose committees of 6 from a population of 10 Americans, 7 Italians, 5 Germans. (Remember that people are always distinguishable, i.e., the Americans are A1, ...A10 etc.)

N(no European nationality left out)

= N(at least one of each European nationality)

= N(at least one I and at least one G)

= total − N(no I or no G)

= total − [N(no I) + N(no G) − N(no I & no G)]

$$= \binom{22}{6} - \left[\binom{15}{6} + \binom{17}{6} - \binom{10}{6} \right]$$

N(no nationality left out)
= N(at least one of each nationality)
= N(atg least one A and at least one I and at least one G)
= total − N(no A or no I or no G)

$$= \text{total} - \left[\begin{array}{l} N(\text{no }A) + N(\text{no }I) + N(\text{no }G) \\ -\ \Big[N(\text{no }A\ \&\ \text{no }I) + N(\text{no }A\ \&\ \text{no }G) + N(\text{no }I\ \&\ \text{no }G)\Big] \\ +\ N(\text{no }A\ \&\ \text{no }G\ \&\ \text{no }I) \end{array}\right]$$

$$= \binom{22}{6} - \left[\binom{12}{6} + \binom{15}{6} + \binom{17}{6} - \left(0 + \binom{7}{6} + \binom{10}{6}\right) + 0 \right]$$

warning

1. Don't try to do "at least one of each nationality" by picking a person from each nationality to be sure. That will double count.

2. The opposite of "at least one of each European nationality" is *not* "no Europeans"; i.e., the opposite is *not* "no I and no G". The correct opposite is "no I OR no G".

some opposite (complementary) events

event	opposite
A or B	not A and not B
A and B	not A or not B
at least one King	no Kings
at least one King and at least one Queen	no Kings or no Queens
at least one King or at least one Queen	no Kings and no Queens
at least one of each suit	no S or no H or no C or no D
all reds in poker	at least one black
at most 3 Blues and at most 4 Greens	at least 4 B or at least 5G

warning

The opposite of "at least one K and at least one Q" is *not* "no K and no Q". The correct opposite is "no K *or* no Q".

example 3 (tricky)

A box contains

100 Reds $R_1, \ldots, R_{100}$

100 Whites $W_1, \ldots, W_{100}$

100 Blacks $B_1, \ldots, B_{100}$

100 Greens $G_1, \ldots, G_{100}$

100 Yellows $Y_1, \ldots, Y_{100}$.

Draw 30 without replacement.
I'll find the number of samples containing (exactly) 3 colors.

wrong way

step 1 Pick which 3 colors to have. Can be done in $\binom{5}{3}$ ways.

Say you pick R, W, B.

step 2 Pick 30 from the 300 R, W, B

Can be done in $\binom{300}{30}$

"Answer" is $\binom{5}{3}\binom{300}{30}$ WRONG

It's wrong because it allows *fewer* than 3 colors. Some of the outcomes counted are

(1) 30 Reds (just one color)
(2) 10 R, 20 B (just two colors)

right way

step 1 (same as above) Pick which 3 colors to have. Can be done in $\binom{5}{3}$ ways.

Say you pick R, W, B.

step 2 Pick 30 from the 300 R, W, B so as to get *at least one of each*.
Use total − opp.

Total is $\binom{300}{30}$

Opp = N(no R or no W or no B)

$= N(\text{no R}) + N(\text{no W}) + N(\text{no B})$ (there are three 1-at-a-time terms)

$+ \Big[N(\text{no R \& no B}) + \text{other 2-at-a-time terms}\Big]$

(there are $\binom{3}{2}$ 2-at-a-time terms

$= 3\binom{200}{30} - \binom{3}{2}\binom{100}{30}$

Final answer is $\binom{5}{3}\Big[\binom{300}{30} - 3\binom{200}{30} + \binom{3}{2}\binom{100}{30}\Big]$

PROBLEMS FOR SECTION 1.7

1. How many poker hands have at least (a) 1 spade (b) 2 spades (c) 4 spades

2. How many bridge hands have at most 2 spade.

3. A hundred people including the Smith family (John, Mary, Bill, Henry) buy a lottery ticket apiece. Three winning tickets will be drawn without replacement. The Smith family will be happy if someone in their family wins. Find the number of ways in which the family can end up happy. (For practice, see if you can find three methods.)

4. Is this a correct way to count the number of 6-letter words containing two or more Z's.
Pick 2 spots for the Z's.
Fill each of the remaining 4 spots with any of 26 letters (i.e., allow more Z's).
Answer is $\binom{6}{2}\cdot 26^4$.

5. (a) How many bridge hands have a void, i.e., are missing at least one suit.
(b) How many bridge hands have at least one royal flush (AKQJ10 in same suit).

6. Pick 11 letters from the alphabet. The order of the draw doesn't matter. How many possibilities have at most 3 vowels if the drawing is
(a) without replacement
(b) with replacement
Note. A committee with 5A and 6Z is considered to have 5 vowels.
A committee with 2A, 3E and 5Z has 5 vowels.

7. In how many ways can 10 cookies be given to 4 people so that no person gets left out if
(a) the cookies are all oatmeal
(b) the cookies are all different

8. (a) Look at strings of length 10 using the letters a, b, c. How many contain
(i) at least one b
(ii) at least one of each letter

(b) Repeat part (a) but with committees of size 10 (necessarily with repeated members) instead of strings.

9. There are 50 states and 2 senators from each state. How many committees of 15 senators can be formed containing at least one from each of Hawaii, Massachusetts and Pennsylvania.

10. How many permutations of the 26 letters of the alphabet contain neither cat nor dog.

11. How many poker hands contain 4 pictures including at least one ace.

12. How many 6-letter words have
(a) A and B appearing once each
(b) A and B appearing at least once each
(c) at least one from the list A, B; i.e., at least one A or at least one B
(d) at least one A and exactly one B
(e) two A's and at least two B's

13. (the game of rencontre---the matching game)
Look at all the ways in which seven husbands $H_1, \ldots, H_7$ and their wives $W_1, \ldots, W_7$ can be matched up to form seven coed couples.
For example, one match is H_1W_3, H_2W_1, H_3W_7, H_4W_4, H_5W_5, H_6W_6, H_7W_2
(a) Find the number of matches in which H_3 is paired with his own wife.
(b) Find the number of matches in which H_2 and H_5 are paired with their own wives.
(c) Find the number of matches in which at least one husband is paired with his wife and simplify to get a pretty answer.
Suggestion: The only feasible method is to use
N(at least one match)
= N(H_1 gets his own wife or H_2 gets his wife or ... or H_7 gets his wife)
(d) Find the number of matches in which no husband is paired with his wife.

14. (like example 3 but this time with 100 identical red balls, 100 identical whites etc)
A box contains
100 Reds (identical)
100 Whites
100 Blacks
100 Greens
100 Yellows

Draw 30 without replacement.
Find the number of samples containing (exactly) 3 colors.

SECTION 1.8 REVIEW PROBLEMS

1. Given a population of 20 people. In how many ways can you
   (1) choose a leader, a senior assistant and a junior assistant
   (2) choose a leader and two assistants
   (3) line them up for a photo if the group includes a set of identical triplets
   (4) choose a committee of 6 which must contain either Tom or Mary
   (5) choose a committee of 6 not containing Sam
   (6) give out 5 different books if a person is allowed to get more than one book
   (7) give out 5 different books if no one can get more than one book
   (8) give out 5 copies of the same book if no one can get more than one copy
   (9) give out 5 copies of the same book if a person can get more than one copy

2. How many poker hands have
   (a) 3 kings
   (b) at least 3 kings
   (c) no clubs
   (d) only clubs
   (e) all one suit
   (f) no suit missing

3. Rewrite the product $67 \cdot 66 \cdot 65 \cdot \ldots \cdot 35 \cdot 34$ more compactly using factorial notation.

4. Start with 5 Men and 12 Women. How many permutations are there
   (a) with the men together
   (b) where the two end positions are men
   (c) with no two men adjacent

5. Given 6 balls all of different colors and 4 boxes $B_1, \ldots, B_4$.
   In how many ways can the balls be distributed among the boxes
   (a) if there are no restrictions
   (b) so that B_2 is empty
   (c) so that B_2 gets at least one ball
   (d) so that no box is empty

6. A jury pool consists of 25 women and 17 men. Among the men, 2 are Hispanic and among the women, 3 are Hispanic.
   A jury of 12 people will be chosen from the pool.
   (a) A jury is called unrepresentative if it contains no women.
   It is also called unrepresentative if it contains no Hispanics.
   How many unrepresentative juries are there.
   (b) A jury is very unrepresentative if it contains neither women nor Hispanics.
   How many very unrepresentative juries are there.

7. A fast food stand sells hamburgers, pizza, tacos, chicken. If a car with 12 people drives up and each person orders one item, how many different orders could the waitress get (e.g., one possibility is 11 pizzas and a taco).

8. In how many ways can the symbols A, B, C, D, 2, 3, 4, 5 be permuted if the numbers and letters must alternate.

9. How many poker hands have
   (a) the ace of spades or KQJ of spades
   (b) all hearts or no hearts

10. (a) John, Mary and Tim sit in an empty row of 10 theater seats.
For instance, some possibilities are

J – – M – – – – – – T
J – – T M – – – – –
J T M – – – – – – –

(a) In how many ways can it be done.
(b) In how many ways can it be done so that they sit together.
(c) In how many ways can it be done so that none of them are together.

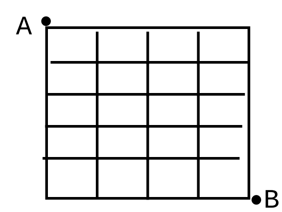
11. In how many ways can 10 people be assigned to 3 identical jobs in a typing pool, 4 identical file clerk jobs and 3 *non*-identical administrative jobs A_1, A_2, A_3.

12. Start with 10 men, 20 women, 30 boys, 40 girls. How many committees are there
(a) of size 4 containing one of each type
(b) of size 7 with 2 men, 4 women and 1 boy
(c) of size 12 with at least 3 women
(d) of size 12 with at most 1 woman

13. Grades will be given in a class of 29 students, S1, ..., S29.
The allowable grades are A, B, C, D, F.
(a) The teacher has to hand in a list of students and their grades.
How many possible lists are there
(b) One possible distribution of grades is 28 A's and 1 B.
Another distribution is 2 A's, 1 B, 23 C's and 3 D's.
How many distributions are there.
(c) Suppose you insist on giving 5 A's, 6 B's, 7 C's, 6 D's and 5 E's. In how many ways can the grades be assigned to the students.

14. The pizza place offers mushroom, anchovies, sausage and onion toppings. You can be as greedy or abstemious as you like. In how many ways can you order your pizza.

15. You want to move from A to B along the streets of the town in the diagram. You are allowed to move only east and south and can change directions at any intersection.
For example one possibility is to walk 1 block E, 5 blocks S and then 3 blocks E.
How many possible routes are there from A to B.



Problem 15

16. Compute (a) $\binom{48}{2}/\binom{48}{3}$ (b) $\binom{100}{98}$

17. Two 5-digit strings (leading zeroes allowed) are considered equivalent if one is a permutation of the other. For example, 12033, 01332, 20331 are equivalent.
How many "distinct" (i.e., non-equivalent) strings are there.

18. You have 26 scrabble chips, one for each letter. How many 5 letter words can be made containing the B chip.

19. A school cafeteria has ten dishes in its repertoire: 3 beef, B_1, B_2, B_3 and 7 vegetarian, $V_1, \ldots, V_7$.

Each day they offer three of the dishes (e.g., B_2, V_4, V_6; B_2, V_4, V_7).

(a) How many days are there in a cycle, before they have to start repeating menus. For example, B_2, V_6, V_7 and B_3, V_6, V_7 is not a menu-repeat but B_2, V_6, V_7 and then B_2, V_6, V_7 again *is* a repeat.

(b) How many times does B_3 appear on the menu in one cycle.

(c) Suppose you only eat beef. On how many days do you get something to eat.

(d) You like to order two lunches, one beef and one veggie. On how many days in a cycle is this possible.

20. In how many ways can four numbers be selected from −5, −4, −3, −2, −1, 1, 2, 3, 4 so that their product is positive if the selection is

(a) without (b) with replacement.

21. Look at permutations of U N U S U A L.

(a) How many are there.
(b) How many have the 3 U's together.
(c) How many have all 4 vowels together.
(d) How many have no consecutive U's.
(e) How many have no consecutive consonants.

22. A hundred messages are sent through 4 available channels. For instance, here are some possibilities

(1) 73 go through C_1, 27 through C_2 and none through C_3 and C_4

(2) 97 go through C_2 and 1 each through C1, C3, C4

etc

(a) How many possibilities are there.

(b) How many possibilities are there in which messages actually go through C_1, C_2 and C_3 but not C_4.

23. A trip is oversubscribed: 15 men, 16 women and 17 teenagers sign up but only 8 can go. In how many ways can the 8 be chosen so that

(a) the group has at least one adult and the same number of men as women.
(b) the group has 2 men and at least 2 women

24. A box contains 260 letters, 10 each of A–Z. Draw 6.

For example, one possibility is 6 A's; another is 2 A's and 4 Q's.

(a) How many possibilities are there.

And does it matter whether the drawing is with or without replacement.

(b) How many of the possibilities in (a) have 3 pairs. For instance 2A, 2Q, 2Z is one possibility. Note that 4B and 2J does *not* count as 3 pairs.

SECTION 1.9 BINOMIAL AND MULTINOMIAL EXPANSIONS

the binomial expansion

When $(x + y)^7$ is multiplied out you can write the result like this:

$$(x+y)^7 = \binom{7}{0}x^7 + \binom{7}{1}x^6 y + \binom{7}{2}x^5 y^2 + \binom{7}{3}x^4 y^3 + \binom{7}{4}x^3 y^4 + \binom{7}{5}x^2 y^5 + \binom{7}{6}xy^6 + \binom{7}{7}y^7$$

In general:

$$\boxed{(x + y)^n = \binom{n}{0} x^n + \binom{n}{1} x^{n-1} y + \binom{n}{2} x^{n-2} y^2 + \ldots + \binom{n}{n-1} xy^{n-1} + \binom{n}{n} y^n} \tag{1}$$

For example, when $(x + y)^9$ is multiplied out, the coefficient of the term x^3y^6 is

$$\binom{9}{6} = \frac{9!}{6!\,3!} = 84$$

why (1) works

Remember that when you expand something like

$$(a + b + c)(d + e)(f + g)$$

each term in the answer comes from multiplying together one term from each of the parentheses.

For instance one term in the expansion is adf (take a from the 1st paren, d from the 2nd, f from the 3rd).

Another term is adg (take a from the 1st paren, d from the 2nd, g from the 3rd).

A *non*-term is abd since a and b come from the same paren.

Now look at

$$(x + y)^7 = (x+y)(x+y)(x+y)(x+y)(x+y)(x+y)(x+y)$$

I'll show that there's a term $x^3 y^4$ in the expansion and its coeff is $\binom{7}{4}$.

One of the terms in the product is x x x y y y y, i.e.,

x from first paren
x from second paren
x from third paren
y from fourth paren
y from fifth paren
y from sixth paren
y from seventh paren.

Another is x y x y y y x, i.e.,

x from first paren
y from second paren
x from third paren
y from fourth paren
y from fifth paren
y from sixth paren
x from seventh paren.

Each of these is x^3y^4 and there are as many of them as there are ways of permuting 3 x's and 4 y's, namely $\frac{7!}{3!\,4!}$.

So there is an $x^3 y^4$ term with coeff $\frac{7!}{3!\,4!}$, or equivalently $\binom{7}{4}$.

Pascal's triangle and Pascal's identity

Fig 1 shows the coeffs in the expansion of $(x + y)^n$ lined up in rows to form *Pascal's triangle*.

Each line of coeffs can be gotten from the preceding line with the indicated addition process

Fig 2 shows the *same* triangle in combinatorial notation.

$$\begin{array}{ccccccccccc}
 & & 1 & & 2 & & 1 & & & & \longleftarrow \text{coeffs for } (x+y)^2 \\
 & 1 & & 3 & & 3 & & 1 & & & \longleftarrow \text{coeffs for } (x+y)^3 \\
1 & & 4 & & 6 & & 4 & & 1 & & \longleftarrow \text{coeffs for } (x+y)^4 \\
 & \text{add} & & \text{add} & & \text{add} & & \text{add} & & & \\
 & \searrow\swarrow & & \searrow\swarrow & & \searrow\swarrow & & \searrow\swarrow & & & \\
1 & 5 & & 10 & & 10 & & 5 & & 1 & \longleftarrow \text{coeffs for } (x+y)^5
\end{array}$$

FIG 1

$$\begin{array}{ccccccccc}
 & & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{2} & & \\
 & \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} & \\
\binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4} \\
 & \text{add} & & \text{add} & & \text{add} & & \text{add} & \\
 & \searrow\swarrow & & \searrow\swarrow & & \searrow\swarrow & & \searrow\swarrow & \\
\binom{5}{0} & \binom{5}{1} & & \binom{5}{2} & & \binom{5}{3} & & \binom{5}{4} & \binom{5}{5}
\end{array}$$

FIG 2

In the notation of Fig 2, the addition rule from Fig 1 looks like this:

$$\binom{4}{0} + \binom{4}{1} = \binom{5}{1}, \qquad \binom{4}{1} + \binom{4}{2} = \binom{5}{2} \qquad \text{etc.}$$

In general:

$$(2) \qquad \boxed{\binom{n}{k-1} + \binom{n}{k} = \binom{n+1}{k} \qquad (\textit{Pascal's identity})}$$

why (2) works

Think of committees of size k chosen from n+1 people $P_1, \ldots, P_{n+1}$. Then

total committees = committees with P_1 + committees without P_1

The total number of committees is $\binom{n+1}{k}$.

There are $\binom{n}{k-1}$ committees containing P_1 and $\binom{n}{k}$ committees not containing P_1 so

$$\binom{n}{k-1} + \binom{n}{k} = \binom{n+1}{k} \qquad \text{QED}$$

example 1

The next line in the triangle in Fig 1 is

1 6 15 20 15 6 1

so

$$(x + y)^6 = x^6 + 6x^5 y + 15x^4 y^2 + 20x^3 y^3 + 15x^2 y^4 + 6xy^5 + y^6$$

example 2

The coeff of $x^{26} y^{31}$ in the expansion of $(x + y)^{57}$ is $\binom{57}{31}$ or equivalently $\binom{57}{26}$

the multinomial expansion

Here's an illustration of the general idea. If

$$(a + b + c + d)^{17}$$

is multiplied out, there are terms of the form

$$ab^5cd^{10}, \quad b^{16}d, \quad a^9b^2c^3d^3 \quad \text{etc.}$$

(In each term, the exponents add up to 17.)

> The coefficient of $a^8b^2c^3d^4$ in the expansion of $(a+b+c+d)^{17}$ is
>
> (3) $$\frac{17!}{8!\,2!\,3!\,4!},$$
>
> called a multinomial coefficient.

why (3) works

When $(a + b + c + d)^{17}$ is multiplied out, each term is created by taking one letter from each of the 17 parentheses.

One of the terms in the product is

a a a a a a a a b b c c c d d d d,

i.e.,

a's from first eight parens
b's from next two parens
c's from next three parens
d's from next four parens

Another is

a b a a a a a a a b c c c d d d d.

Each of these is $a^8b^2c^3d^4$ and there are as many of them as there are ways of permuting 8 a's, 2 b's, 3 c's, 4 d's. So there is an $a^8b^2c^3d^4$ term with coeff $\frac{17!}{8!\,2!\,3!\,4!}$ ((1) inSection 1.4)

example 3

In the expansion of $(x + y + z)^{15}$

the coeff of $x^4y^5z^6$ is $\frac{15!}{4!\,5!\,6!}$

the coeff of x^5z^{10} is $\frac{15!}{5!\,0!\,10!}$ (you can leave out the 0! since it's 1)

etc.

PROBLEMS FOR SECTION 1.9

1. Find the coeff of $x^{43}y^6$ in the expansion of $(x + y)^{49}$.

2. Expand $(x + y)^7$ using Pascal's triangle and check some of the coeffs with (1).

3. Find the coeff of (a) $a^2bc^4d^6$ (b) $abcd^{10}$ in the expansion of $(a+b+c+d)^{13}$.

4. Find the coeff of $x_3^5\, x_4^8$ in the expansion of $(x_1 + x_2 + x_3 + x_4)^{13}$.

5. How many terms are there in the expansion of $(x + y + z + w)^{17}$ (after you combine like terms).

Suggestion: A typical term is of the form $x^{\square}\, y^{\square}\, z^{\square}\, w^{\square}$ where the sum of the boxes is 17.

6. The last problem in Section 1.6 was solved twice. One method gave the answer $\binom{50}{4} + \binom{50}{4}$ and a second method gave the answer $\binom{51}{4} + \binom{51}{4} - 2\binom{50}{3}$.
Use Pascal to reconcile the two answers without resorting to any numerical computation.

SECTION 1.10 PARTITIONS

dividing people into distinguishable groups

Here's an example to illustrate the general idea.

Suppose a group of 30 people is to be divided (partitioned) into a

4-person school board
4-person park board
4-person sewer board
5-person street board
5-person tree board
7-person housing board
1-person transportation board

To find the number of possibilities, use each board as a slot:

(1) $$\text{Answer} = \binom{30}{4}\binom{26}{4}\binom{22}{4}\binom{18}{5}\binom{13}{5}\binom{8}{7}$$

footnote

The expression in (1) is

$$\frac{30!}{4!\,26!}\;\frac{26!}{4!\,22!}\;\frac{22!}{4!\,18!}\;\frac{18!}{5!\,13!}\;\frac{13!}{5!\,8!}\;\frac{8!}{7!\,1!}$$

which cancels down to the more compact version

(2) $$\frac{30!}{(4!)^3\,(5!)^2\,7!\,1!}$$

In this section I mostly stick with the longer form.

more footnote

Here's another method which jumps right to the compact form in (2).

Print up the following 30 labels

4 school board labels
4 park board labels
4 sewer board labels
5 street board labels
5 tree board labels
7 housing board labels
1 transportation board label

Permute the labels. Can be done in $\dfrac{30!}{4!\;4!\;4!\;5!\;5!\;7!\;1!}$ ways by (1) in §1.4

Now that the labels are permuted, pin them respectively on the people P1,..., P30. (one way to do it).

So the final answer is the one in (2).

dividing people into indistinguishable (nameless) groups

Take the same 30 people again but this time find the number of ways to divide (partition) them into seven groups (cells) of sizes 4,4,4,5,5,7,1.

For example, the list of outcomes looks like this:

outcome 1

P1, P2, P3, P4
P5, P6, P7, P8
P9, P10, P11, P30
P12–P16
P17–P21
P22–P28
P29

outcome 2

```
P1, P2, P3, P29
P5, P6, P7, P8
P9, P10, P11, P30
P12-P16
P17-P21
P22-P28
P4
```

etc.

To count the number of outcomes here, you can't use slots because there is no such thing as the *first* group of size 4, the *second* group of size 4, the *third* group of size 4 and similarly the two groups of size 5 aren't distinguished from one another. So this problem is different from the one above. Here's how to get an answer using the connection between this problem and the previous one.

Each possibility in *this* problem gives rise to $3!\,2!$ possibilities in the *preceding* example because once you have divided the 30 people into groups of sizes 4,4,4,5,5,7,1 you can give the *three* groups of size 4 the labels school, park, sewer in $3!$ ways; and you can give the *two* groups of size 5 the labels street, tree in $2!$ ways. For example, item 1 above gives rise to the following possibilities in the *preceding* example:

outcome 1a

P1, P2, P3, P4	school
P5, P6, P7, P8	park
P9, P10, P11, P30	sewer
P12-P16	street
P17-P21	tree
P22-P28	housing
P29	trans

outcome 1b

P1, P2, P3, P4	park
P5, P6, P7, P8	school
P9, P10, P11, P30	sewer
P12-P16	street
P17-P21	tree
P22-P28	housing
P29	trans

etc

So

answer to this problem $\times\ 3!\,2!$ = answer to the preceding problem

$$\text{answer to this problem} = \frac{\text{answer to preceding problem}}{3!\;2!}$$

The number of ways to divide 30 people into groups of sizes 4,4,4,5,5,7,1 is

$$\frac{\binom{30}{4}\binom{26}{4}\binom{22}{4}\binom{18}{5}\binom{13}{5}\binom{8}{7}}{3!\;\;2!} \quad \text{<--- extra factors}$$

example 1

Start with 12 people.
In how many ways can they be divided into four trios.

solution

I'm going to do another problem first: Divide the 12 people into trios named T1, T2, T3, T4 (say to cook, serve, wash, dry). For this new problem you can use slots:

$$\text{answer to new problem} = \binom{12}{3}\binom{9}{3}\binom{6}{3}$$

In the original problem, the trios do not have names.

Answer to original problem × 4! ways to name the trios = answer to new problem.

$$\text{So the answer to the old problem} = \frac{\text{answer to new problem}}{4\,!}$$

$$= \frac{\binom{12}{3}\binom{9}{3}\binom{6}{3}}{4\,!} \quad \text{<--- extra factor}$$

clarification

The answer to example 1 is *not* $\binom{12}{3}\binom{9}{3}\binom{6}{3}$. This "answer" double counts (by a factor of 4!). For example, it counts the following outcomes as different when they are really the same.

outcome 1 $P_1, P_2, P_3;\ \ P_4, P_5, P_6;\ \ P_7, P_8, P_9;\ \ P_{10}, P_{11}, P_{12}$

outcome 2 $P_4, P_5, P_6;\ \ P_1, P_2, P_3;\ \ P_7, P_8, P_9;\ \ P_{10}, P_{11}, P_{12}$

example 2

(a) In how many ways can 76 people be divided into 9 cells of sizes 7, 7, 13, 13, 13, 13, 6, 3, 1.

(b) In how many ways can 76 people be divided among 9 hotels $H_1, \ldots, H_8$ if

H_1 and H_2 have 7 vacancies each

H_3, H_4 H_5 and H_9 have 13 vacancies each

H_6 has 6 vacancies

H_7 has 3 vacancies

H_8 has 1 vacancy

solution

(a) The two groups of size 7 are not distinguishable from one another and neither are the four groups of size 13 so if you use 9 slots you'll have to divide by the extra factors 2!4! to compensate. The answer is

$$\frac{\binom{76}{7}\binom{69}{7}\binom{62}{13}\binom{49}{13}\binom{36}{13}\binom{23}{13}\binom{10}{6}\binom{4}{3}}{2!\,4!}$$

(b) Use the hotels as the slots:

$$\binom{76}{7}\binom{69}{7}\binom{62}{13}\binom{49}{13}\binom{36}{13}\binom{23}{13}\binom{10}{6}\binom{4}{3}$$

footnote (another point of view)

The problem in (b) can be thought of as tossing 76 distinguishable balls into 8 distinguishable boxes $B_1, \ldots, B_9$ so that B_1 and B_2 get 7 balls each, B_3, B_4, B_5, B_9 get 13 balls each, B_6 gets 6 balls, B_7 gets 3 balls and B_8 gets 1 ball.

The problem in part (a) is to toss 76 distinguishable balls into 9 *indistinguishable* boxes so that 2 of the boxes get 7 balls each, 4 boxes get 13 balls each, one box gets 6 balls, one box gets 3 balls and one box gets 1 ball.

example 3 (when the two types coincide)

(a) In how many ways can 12 people be divided into groups of sizes 3, 4, 5.
(b) In how many ways can 12 people be divided into a 3-person school board, a 4-person park board and a 5-person sewer board.

solution

These are the *same* problem. In part (a), the groups are named "group of 3", "group of 4" and "group of 5". In each case the answer is

$$\binom{12}{3}\binom{9}{4}$$

For part (b) the slots are school, park, sewer.
For part (a) the slots are "group of 3", "group of 4", "group of 5".

example 4

In how many ways can you divide 30 people into two trios, five quartets and two duos named D1 and D2.

solution

First do this problem. Divide 30 people into two trios T1, T2, five quartets Q1, ... Q5 and the duos D1, D2. Use the T's, Q's and D's as slots. Can be done in

$$\binom{30}{3}\binom{27}{3}\binom{24}{4}\binom{20}{4}\binom{16}{4}\binom{12}{4}\binom{8}{4}\binom{4}{2}$$

In the original problem the trios were indistinguishable and the quartets were indistinguishable. Each outcome in the original problem gives rise to $2!5!$ outcomes in the new problem (the trios can be named in $2!$ ways, the quartets in $5!$ ways). So

$$\text{answer to original problem} \times 2!\,5! = \text{answer to new problem}$$

$$\text{answer to original problem} = \frac{\binom{30}{3}\binom{27}{3}\binom{24}{4}\binom{20}{4}\binom{16}{4}\binom{12}{4}\binom{8}{4}\binom{4}{2}}{2!5!}$$

PROBLEMS FOR SECTION 1.10

1. Four people will divide up 16 books B1,..., B16 so that Tom and Dick get 6 apiece, Harry gets 1 and Mary gets 3. In how many ways can it be done.

2. (a) Twelve people at a party divide into four conversational groups of 3 each. In how many ways can it be done.

(b) Twelve office workers arrange their vacations so that 3 go in each of the months of June, July, Aug, Sept. How many vacation schedules are there.

3. There are 100 children in the fifth grade.
(a) In how many ways can they be split into 4 sections of 25 each.
(b) In how many ways can 25 be assigned to Ms. X as their teacher, 25 to Mr. Y, 25 to Z and 25 to W.
(c) In how many ways can they be split into 4 sections of 25 each so that John and Mary are in different sections
(d) In how many ways can they be split into 4 sections of 25 each so that John and Mary are in the same section.

4. Twenty-one people show up for a tennis tournament. In how many ways can the 10 matches and one bye be arranged. Cancel to get a short answer.

5. In how many ways can 18 players be divided into
   (a) two teams of 9 each
   (b) a home team of size 9 and an away team of size 9

6. In how many ways can 170 people be divided into

2 groups of size 5
3 groups of size 20
4 groups of size 10
a red team of size 30
a blue team of size 30

Cancel to get a compact answer.

7. 350 people register for a course.
   (a) In how many ways can they be split into sections of sizes 50, 100, 200.
   (b) In how many ways can they be split into 50 for Section A, 100 for Section B and 200 for Section C.
   (c) In how many ways can they be split into sections of sizes 50, 100, 200 so that John and Mary are in the same section.

8. A company makes 20 different cereals. They decide to market them in a 10-pack and two 5-packs. They have to decide how to divide up the cereals among the packs. For instance should Tweeties go in the 10-pack and if so with what other 9 cereals or should Tweeties go in a 5-pack and if so with what 4 other cereals. How many possibilities are there.

9. Here are two problems with proposed solutions. Are the solutions correct. If a solution is not correct, explain why not and fix it.

(a) To count the ways to divide 100 people into two groups of size 50:

Pick 50 of the 100 people. Can be done in $\binom{100}{50}$ ways.

This automatically leaves a second group of 50.

Answer is $\binom{100}{50}$.

(b) To counts the ways to divide 100 people into groups of size 25 and 75:

Pick 25 of the 100 people. Can be done in $\binom{100}{25}$ ways.

This automatically leaves a group of size 75.

Answer is $\binom{100}{25}$.

REVIEW PROBLEMS FOR CHAPTER 1

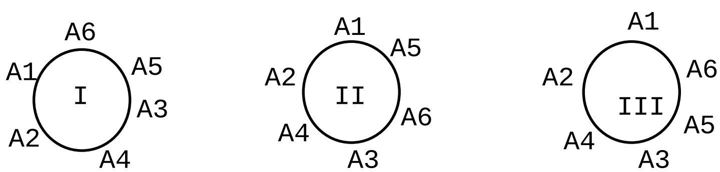
1. Show that $\binom{r-1}{r-n} = \binom{r-1}{n-1}$

2. Consider strings of digits of length 7 (e.g., 0000000, 2345678).
 (a) How many are there.
 (b) How many contain no 5's.
 (c) How many have at least one 2.
 (d) How many have an least one 2 or 3.
 (e) How many have at least one 2 and at least one 3.
 (f) how many have one 3 and two 4's.
 (g) how many have one 3 and at least two 4's.

3. From a pool of 50 people, choose 5 for a basketball game, 4 for a hockey game and 9 for a baseball game. In how many ways can it be done if
 (a) the 3 games are played simultaneously
 (b) the games are played on different days

4. Consider 6-letter words using only the three letters P,Q,R.
 (a) How many have 4 of one kind, one of a second kind and one of a third kind (e.g., RRRRPQ, PPQPRP).
 (b) How many have 3 of a kind and a pair (e.g., PPPRRQ, QRQRQP).

5. There are 6! ways to arrange $A_1, \ldots, A_6$ on a *line*. In how many ways can they be arranged on a *circle*.
 Note that circles I and II below are different but I and III are the same (spinning doesn't change the circle).



6. (a) Find the coeff of $x^9 y^{11}$ in the expansion of $(x + y)^{20}$.

 (b) Find the coeff of $a^2 b^3 c^4$ in the expansion of $(a + 5b + c)^9$.

7. You want to count the ways to divide 10 people into 5 pairs.
 (a) Explain why these proposed solutions are wrong.
 (i) There are $\binom{10}{2} = 45$ pairs. Pick 5 of them. Answer is $\binom{45}{5}$ WRONG
 (ii) Pick 5 people. Then match these 5 with the remaining 5.
 Answer is $\binom{10}{5}\, 5!$ WRONG
 (iii) Keep picking twosomes: $\binom{10}{2}\binom{8}{2}\binom{6}{2}\binom{4}{2}\binom{2}{2}$ WRONG
 (b) Find the right answer.

8. Four companies $C_1, \ldots, C_4$ bid for eleven different government grants. In how many ways can the grants be awarded if C_2 must get between 2 and 5 grants.

9. How many permutations of the 26 letters of the alphabet have exactly three letters between A and B.

10. Show that $\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \ldots + \binom{n}{n} = 2^n$

(a) by expanding $(1 + 1)^n$

(b) with a combinatorial proof by making up a counting problem and solving it two different ways.

11. You have 10 hours to study for History, Math, Spanish. For example two possibilities are

9 hours on H, 1 hour on S

3 hours on H, 3 hours on M, 4 hours on S

How many possibilities are there.

12. You are required to take

one course from A_1, A_2, A_3

two courses from B_1, B_2, B_3, B_4, B_5

one course from C_2, C_2, C_3, C_4

In how many ways can you fill the requirements.

13. In how many ways can 20 offices be linked by intercoms if there is equipment for three 2-office hookups, one 4-office hookup and two 5-offices hookups.

14. Use Pascal's identity to combine $\binom{n}{k} + 2\binom{n}{k-1} + \binom{n}{k-2}$ into a single binomial coefficient.

15. You're spending 7 days at a resort and there are four activities A, B, C, D available.

(a) Suppose you want to do one activity each day so as to eventually try each one. For example, one schedule might be

Monday	B
Tues	B
Wed	A
Thurs	D
Fri	C
Sat	C
Sun	B

How many of these schedules are there.

(b) Suppose you are going to rest for four days and do a different activity on each of the other 3 days. For example, one schedule might be

Mon	rest
Tues	rest
Wed	B
Thurs	rest
Fri	A
Sat	rest
Sun	C

How many of these schedules are there.

16. Use algebra to simplify $\binom{2n+3}{2n+1} - 3\binom{n+2}{n}$.

17. A store has 7 A's, 4 B's and 2 C's in stock. You come in to buy eight items. But if you ordered 1 A, 5 B's and 2 C's the store couldn't fill your order because it doesn't have enough B's. How many unfillable orders are there.

18. How many strings of 0's and 1's of length 7 have three 1's.

19. Five people $P_1, \ldots, P_5$ sit down in a 12 chair row. For example here's one possibility:

$$\_\_\ P_5\ \_\_\ P_1\ P_2\ \_\_\ P_4\ \_\_\ \_\_\ \_\_\ P_3\ \_\_$$

Find the number of possibilities using the following methods.
(a) Fill slots.
(b) Permute 5 people and 7 empty chairs.
(c) Pick chairs for the people and then fill them.
(d) (excruciatingly clever) Line up the people and then toss 7 identical chairs into the 6 in-betweens and ends.

20. The guidebook to Rome recommends 13 churches and 4 museums.
One possibility is that you visit C_1, C_5, C_{12}, M_2.
Another possibility is that you visit only C_2.
How many possibilities are there.

21. How many 4 letter words have no repeats.
For example, ABAC is no good; PQBC is OK.

22. In how many ways can 10 marbles be put into 4 distinct containers if
(a) all the marbles look the same
(b) the marbles are all different colors
(c) the marbles are all the same and each box must get at least one marble
(d) the marbles are all different colors and each box must get at least one marble

23. How many permutations of the alphabet contain
(a) MOTHER but not PA.
(b) neither MOTHER nor PA

24. Out of 100 students, 20 will be picked for academic honors and 12 for athletic honors (overlap allowed, i.e., athletic scholars are possible).
(a) In how many ways can it be done.
(b) How many of the possibilities from part (a) have 5 overlaps, i.e., how many of the possibilities give 5 people both athletic and academic honors.

25. A function f(x,y) has the following 2nd order partial derivatives:
f_{xx} (differentiate twice w.r.t. x)
f_{yy} (differentiate twice w.r.t. y)
f_{xy} (differentiate first w.r.t. x and then w.r.t. y)
f_{yx} (differentiate first w.r.t y and then w.r.t. x)

Similarly, the 10th order partials of f(x,y,z,w) are $f_{xxxyzzwwxy}$, $f_{yyyyyyzzzz}$ etc.
(a) How many 10th order partials does f(x,y,z,w) have.
(b) It's a theorem in calculus that $f_{xy} = f_{yx}$. Similarly $f_{xyzz} = f_{zxyz} = f_{yzzx}$ etc. All that counts is how *often* the differentiation is done for each variable; the *order* in which it's done doesn't matter.
With this in mind, how many truly *different* 10th order partials does f(x,y,z,w) have.

26. (a) How many 15-letter words have five 3-of-a-kinds
(e.g., AAABBBCCCQQQZZZ, ABCQZABCQZABCQZ)

(b) How many 15 card hands have five 3-of-a-kinds
(e.g., A_H, A_S, A_C, Q_H, Q_D, Q_S, 3_D, 3_C, 3_S, 9_D, 9_C, 9_S, 7_H, 7_C, 7_S)

CHAPTER 2 GRAPHS

SECTION 1 INTRODUCTION

basic terminology

A graph is a set of finitely many points called *vertices* which may be connected by *edges*. Figs 1-3 show three assorted graphs.

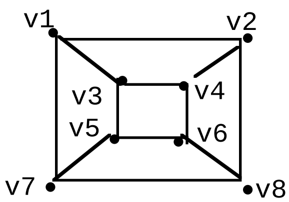


FIG 1

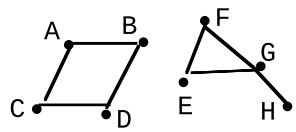


FIG 2

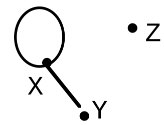


FIG 3

If there is more than one edge between two vertices (Fig 4) then the graph is said to have *multiple edges* or *parallel edges.*

An edge drawn from a vertex back to the same vertex is called a *loop*. See Fig 5 and also vertex X in Fig 3.

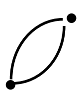


FIG 4
multiple edges

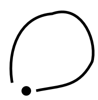


FIG 5
loop

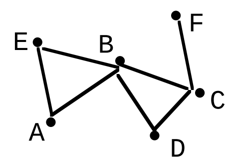


FIG 6

The number of edges connected to a vertex is called the *degree* of the vertex. In Fig 1, all vertices have degree 3; in Fig 3, Y has degree 1, Z has degree 0 (it's an *isolated* vertex). A loop is considered to contribute 2 to the degree of a vertex since the edge leaves the vertex and also returns. In Fig 3, X has degree 3.

If the number of edges is E then the sum of the degrees of the vertices is 2E.

This holds because each edge contributes 2 to the total degree count.

A *path* is a sequence of connected edges (along which you can take a walk through the graph). By convention, a path can repeat a vertex (i.e., a path can cross itself) but can't repeat an edge (no backtracking). In Fig 6, ABCF is a path between A and F; other paths between A and F are AEBCF., ABDCF etc. Some paths between A and E are AE, ABE, ABCDBE (which crosses itself at B).

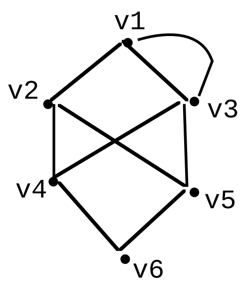
A *connected* graph is one in which there is a path between any two different vertices (i.e., it's always possible to get from here to there). Figs 1,4,5,6 are connected. Fig 2 is not connected since it's impossible to get from any of A,B,C,D to any of E,F,G,H. Fig 3 is not connected because it's impossible to get to Z from X and Y.

The graph in Fig 2 has two *components*, i.e., the graph is split into two connected pieces with no edges between the pieces. Fig 3 also has two components.

A path that ends up where it started is called a *cycle* or a circuit. In Fig 6, ABEA is a cycle; so is the figure eight A B C D B E A. A cycle can repeat a vertex (in fact it must repeat the first and last vertex) but not an edge. A graph with no cycles is *acyclic.*

the adjacency matrix

A graph can be represented by an *adjacency matrix* where each entry indicates how many edges there are between vertices. Fig 7 shows a graph and the corresponding adjacency matrix. The entry in row 3, col 4 is 1 because vertices v3 and v4 are connected by one edge. The entries along the diagonal are 0 because no vertex in the graph has a loop. The entry in row 1, col 3 is 2 because there are two edges between v_1 and v_3.



	1	2	3	4	5	6
1	0	1	2	0	0	0
2	1	0	0	1	1	0
3	2	0	0	1	1	0
4	0	1	1	0	0	1
5	0	1	1	0	0	1
6	0	0	0	1	1	0

diagonal

FIG 7

The adjacency matrix of a graph is *symmetric*, i.e., the entry in row i, col j is the same as the entry in row j, col i since both entries describe the number of edges between vertices v_i and v_j; equivalently, row i and col i are identical.

isomorphic graphs

Given an adjacency matrix, there are many ways in which the corresponding graph can be drawn. For example, the adjacency matrix

	A	B	C	D
A	0	1	0	0
B		0	1	1
C			0	1
D				0

goes with all the graphs in Fig 9. The graphs in Fig 9 are called *isomorphic* since they have the same adjacency matrix.

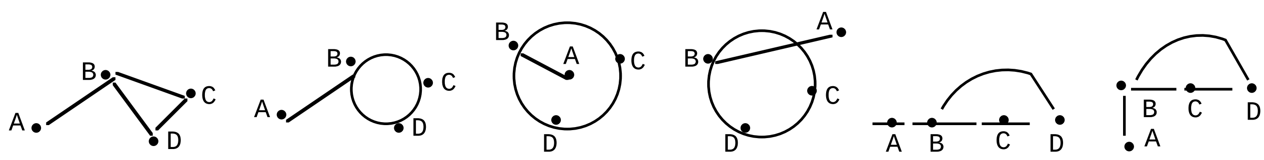


FIG 9 six isomorphic graphs

More generally, graphs don't have to be lettered in the same way to be called isomorphic.

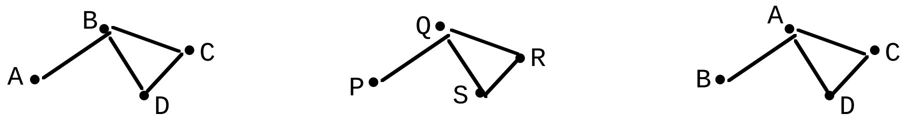


FIG 10 three isomorphic graphs

In Fig 10, the second graph is isomorphic to the first with

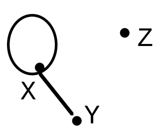
P corresponding to	Q
Q	B
R	C
S	D

And the third graph is also isomorphic to the first, with B playing the role of A and A playing the role of B.

In general: *two graphs are isomorphic if you can match up the letters so that the two adjacency matrices are the same*.

PROBLEMS FOR SECTION 2.1

1. Find the adjacency matrix for this graph



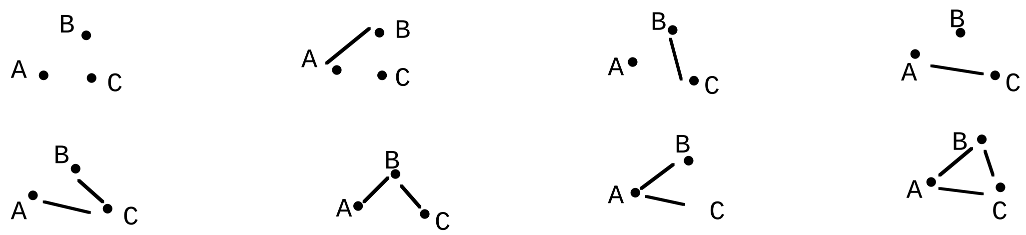
2. How can an isolated vertex (degree 0) be identified by looking at the adjacency matrix.

3. (a) Let S be the sum of the degrees of all the vertices in a graph.
Show that S is always even.
Suggestion. Look at the connection between S and the number of edges.

(b) Use part (a) (among other things) to show that in any graph, there must be an even number of vertices of odd degree.
Suggestion: Start like this.
Let $V_1, \ldots, V_k$ be the vertices of even degree.
Let $Q_1, \ldots, Q_n$ be the vertices of odd degree. (You want to show that n is even.)
Let S be the sum of the degrees as in part (a).

(c) Use part (a) to show that in a group of 25 quarrelsome people it isn't possible for each person to get along with exactly 5 others (assuming that getting along is a 2-way street so that if A gets along with B then B gets along with A).

4. (a good counting problem) To warm up, look at graphs with vertices A, B, C and with no loops or multiple edges.
There are 8 of them:

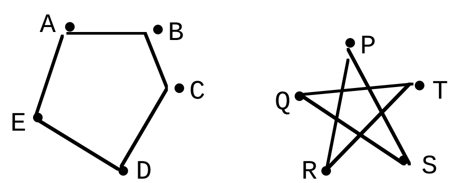


(Yes the graph with 3 vertices and no edges counts.)
(The second, third and fourth graphs are isomorphic but for this problem I'll count them as different.)

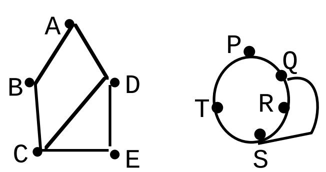
Now consider graphs with ten vertices $A_1, \ldots, A_{10}$.

(a) How many are there with no loops and no multiple edges.
(b) How many are there with no loops but now allowing as many as 2 edges between each pair of vertices
(c) How many are there with no multiple edges but now allowing loops, no more than one loop per vertex.

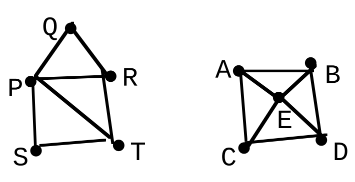
5. For each pair of graphs, decide if they are isomorphic or not.
(a)



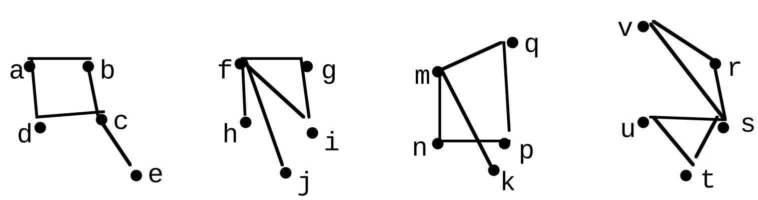
(b)



(c)



6. Are there any pairs of isomorphic graphs among the four shown below.



SECTION 2.2 EULER CYCLES AND EULER PATHS

Euler cycles

A cycle which includes every edge of the graph (exactly once) is called an Euler cycle.

In other words, with an Euler cycle you can walk through the graph hitting every edge exactly once and end up back where you started.

criterion for the existence of an Euler cycle

Consider a connected graph (there's no hope for a non-connected graph).
It has an Euler cycle iff only if the degree of each vertex is even.
In other words

(1) If one or more vertex has odd degree then there is *no* Euler cycle. (2) If all the vertices have even degree then there *is* an Euler cycle.

To see why (1) holds consider the graph in Fig 1 where vertex Z has degree 3. To get an Euler cycle you'd have to include edges e1, e2 and e3.

If you arrive at Z say on e1, leave on say e2, and eventually arrive again on e3 then you can't leave Z; there are no more exits so you're stuck and can't continue the cycle.

So a vertex of degree 3, or, more generally a vertex of odd degree, prevents Euler cycles.

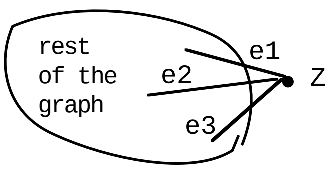


FIG 1

The next algorithm will show that (2) holds since it actually finds an Euler cycle if all the vertices have even degree.

breakout algorithm

Suppose a graph is connected and all vertices have even degree. I'll illustrate how to find an Euler cycle using the graph in Fig 2.
Pick any vertex to start.
Begin traveling along successive edges, never repeating an edge.
Whenever a choice is available, pick the alphabetically or numerically first vertex to be systematic.
With this procedure, starting from vertex A in Fig 2 you get

A B C A stuck

You're stuck because all the edges at A have been used; there are no more exits available.

Note that you're stuck *back at the initial vertex A again*. You can't get stuck anywhere else because every vertex has even degree so there is an exit for every entrance.

So far you have the cycle ABCA (Fig 3) but it doesn't include all edges yet and there is no place to go from A. In this case, backtrack from the exit-less A until you find a vertex that *does* have exits, C in this case, and break out (there must be a way to break out to the edges not included yet since the graph is connected).

From C you have the side excursion C D B E C which starts at C and gets stuck back at C so you can pick up where you left off:

```
        D B E C stuck
A B C ^ A
```

Note that when you break out, the "remaining" degree at each vertex is even so on the side trips you always get stuck back where you broke out. This is what makes the algorithm work.
So far you have the cycle ABCDBECA in Fig 4 but not all edges are included yet so backtrack from the latest stuck position at C and find a place from which to break out, namely from E.

```
                D F E stuck
          D B E ^ C
  A B C ^ A
```

Now there's nowhere to break out meaning that you have traversed all the edges and you're finished. An Euler cycle is

A B C D B E D F E C A (Fig 5)

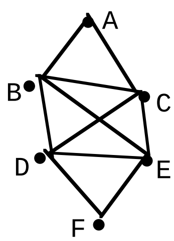


FIG 2

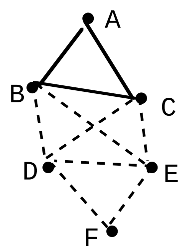


FIG 3

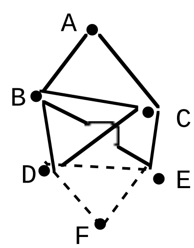


FIG 4

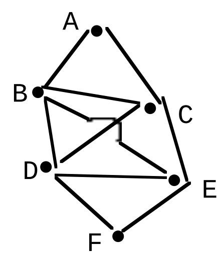


FIG 5

example 1

You might get different Euler cycles by starting at different vertices. In Fig 2, starting from C you get

```
                  D F E
C A B C D B E ^ C
      ↑
```

warning when you get back to C here, do not back up and break out because you are not *stuck* at C. There are still exits available from C. You only break out when you are back where you started and *stuck*.

This can be written neatly as

C A B C D B E D F E C

Starting from F you get F D B A C B E C D E F, no breakout necessary.
The cycles starting from C and A respectively happen to be the same but the one you get from F is different. A graph can have many Euler cycles.

warning

To follow the spirit of this algorithm, remember to pick the first alphabetically of all vertices available.

Euler paths

An Euler path is a path which includes all edges (once each) but isn't a cycle (i.e., different starting and stopping points).

Let G be a connected graph. Then *G has an Euler path iff G has precisely two vertices of odd degree* (one to serve as the initial point with one more exit than entrance and one to serve as the end point with one more entrance than exit).

The breakout algorithm can be used to find an Euler path if you start at one of the vertices of odd degree. For example, look at the graph in Fig 6 where B and F have odd degree and the other vertices have even degree.

If you begin say with F you get

```
                 F I H D G H
        F C B A D E ^ B
```

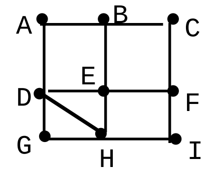


FIG 6

so an Euler path is F C B A D E F I H D G H E B.

Note that when you start from one of the odd degree vertices you'll always get stuck at the other one: you can't get stuck at the initial vertex since it always has one more exit than entrance and you can't get stuck at one of the even-degree vertices since they always have as many exits as entrances.

And when you break out, the remaining degree at each vertex is even so on the side trips you always get stuck back where you broke out.

This makes the algorithm work.

example 2

The graph in Fig 7 has four vertices of odd degree (A,B,C,D).

So it doesn't have an Euler cycle or an Euler path.

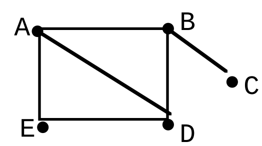


FIG 7

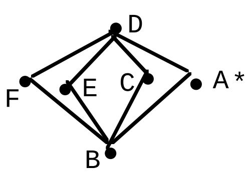
warning

For small graphs it may be easy to find an Euler cycle by inspection. But the problems and the exams want you to use the breakout algorithm because it works automatically, even in large graphs that defy inspection.

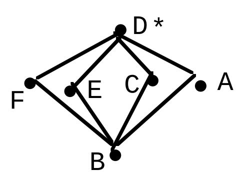
PROBLEMS FOR SECTION 2.2

1. Each graph in the diagram is connected and all vertices have even degree. Find an Euler cycle starting from the starred vertex.

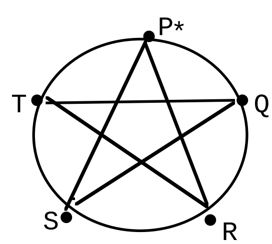
(a)



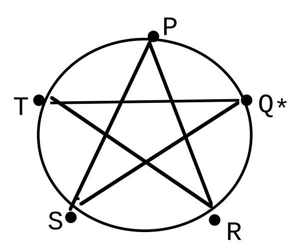
(b)



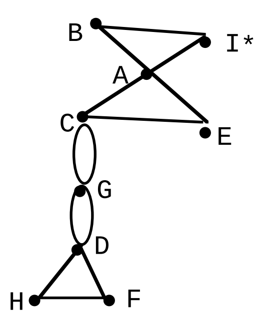
(c)



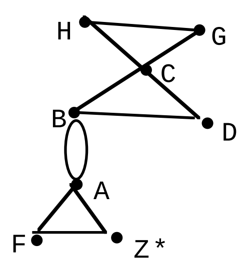
(d)



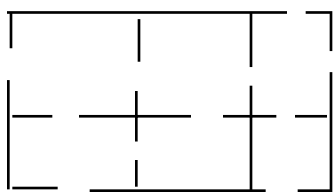
(e)



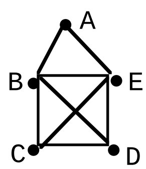
(f)



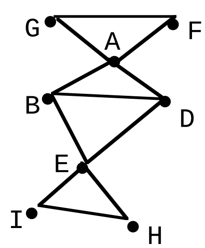
2. With the given floor plan (gaps represent doors), starting from outside the house, is it possible to walk through each interior and exterior door exactly once and end up outside again.
  Suggestion: Represent each room by a vertex, let the great outdoors (the "outside" room) be another vertex, Q, and draw an edge for each door between rooms.



3. Is it possible to draw the figure below without lifting the pencil from the paper and without retracing any lines. If not, explain why not. If possible, explain why you are sure it is possible and then do it.



4. Find an Euler cycle or an Euler path in the graph below if one exists.



SECTION 2.3 SPANNING TREES

definition of a tree

A graph which is *connected and acyclic* (no cycles) is called a tree. Fig 1 shows some trees and Fig 2 shows some non-trees.

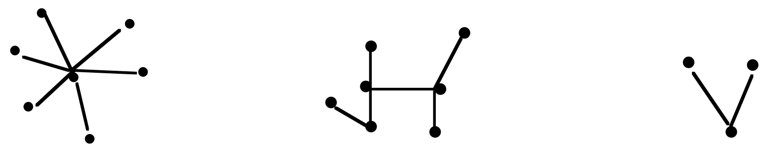
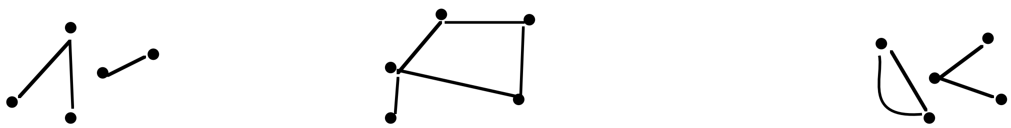


FIG 1 Three trees



Not connected   Has a cycle   Not connected and has a cycle

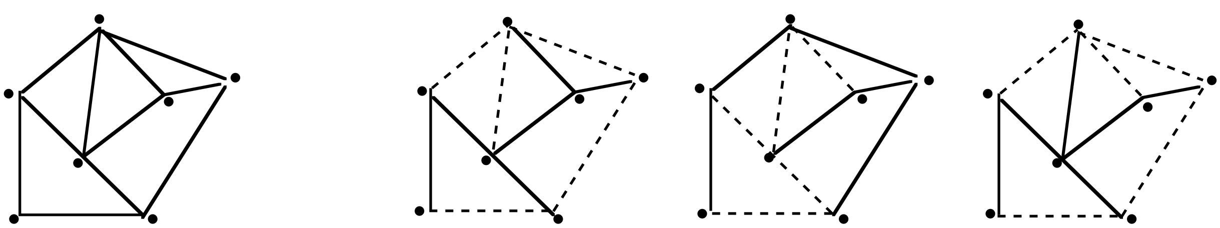
FIG 2 Three *non*-trees

spanning trees

Let G be a connected graph. A spanning tree for G is a tree, built with edges from G and connecting all the vertices of G. Think of it as a skeleton for G, containing all of G's vertices but usually with fewer edges (since G may have cycles but a skeleton doesn't).

Every connected graph has a spanning tree (usually more than one). If G is already a tree then G is its own spanning tree; otherwise a spanning tree for G has fewer edges than G. Fig 3 shows a connected graph (which is not a tree) along with several of its spanning trees.

A graph that isn't connected doesn't have a spanning tree.



graph G   some spanning trees for G

FIG 3

a spanning tree algorithm

There are two popular ways to construct a spanning tree for a connected graph starting from an arbitrary initial vertex.

breadth-first Add *all* the vertices directly connected by an edge to the initial vertex. Then for *each* of those newly added vertices in (alphabetical) turn, add *all* vertices directly connected that are not yet in the tree. And so on.

depth-first Add *one* vertex directly connected by an edge to the initial vertex. If there is more than one possibility, choose the first alphabetically to be systematic.

Then to that newly added vertex add *one* vertex (the first alphabetically) directly connected that is not yet in the tree. And so on.

If there is no way to grow the tree from the *last* vertex added, backtrack to the next-to-last and grow again. If necessary backtrack all the way to the initial vertex.

The breadth-first search grows a spanning tree by sending as many tentacles as possible out through the graph simultaneously. The depth-first search grows a spanning tree by sending one tentacle out through the graph.

I'll illustrate the algorithm by constructing some spanning trees for the graph in Fig 4.

example 1

Here's a *breadth-first* search starting from A in Fig 4.

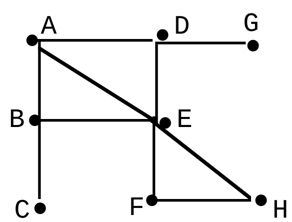
round 1 Begin with A. Each of B,D,E is connected to A so add them to the tree (Fig 5 shows two versions of the spanning tree).

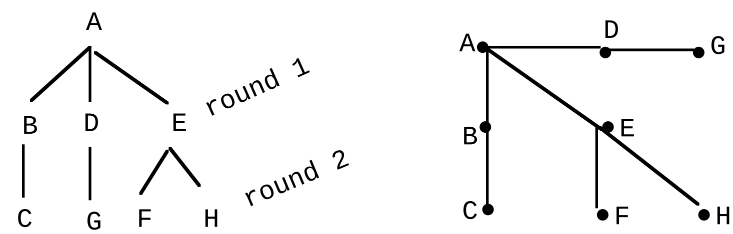
round 2 Look at B. Vertex C isn't in the tree yet and is connected to B so add C.

Look at D. Vertex G is not yet in the tree and is connected to D so add G.

Look at E. Vertices F and H are not yet in the tree and are connected to E so add them.

Now that all the vertices are in the tree, stop. Fig 5 shows two versions of the spanning tree.





Spanning tree for graph in Fig 4

Fig 4

FIG 5

example 2

Fig 6 shows the breadth-first tree (two versions) starting from E in Fig 4.

Fig 7 shows the breadth-first search starting from F in Fig 4

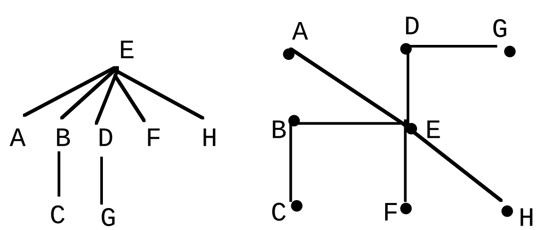


FIG 6

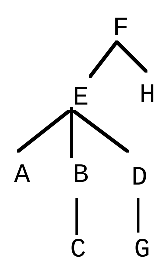


FIG 7

example 3

Here's a *depth-first* search starting from A in Fig 4 (repeated below).

round 1 Start at A. Each of B,D,E is connected to A. Pick *one* of them, say B, the first alphabetically, and add it to the tree (Fig 8).

Now start at B and look at all the vertices, C and E, connected to B but not in the tree yet. Pick one of them, say C, the first alphabetically, and add it to the tree.

Now start at C. None of the remaining vertices is connected to C so backtrack to B and look again.

> **warning**
> My version of the algorithm backtracks to the *latest pick*, B in this case, not A. Do it that way so that we'll all get the same answer.

round 2 From B add E, then D and then G. You're stuck at G now so backtrack to E
round 3 From E add F and then H and you're finished (Fig 8).

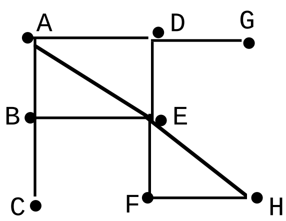


FIG 4 again

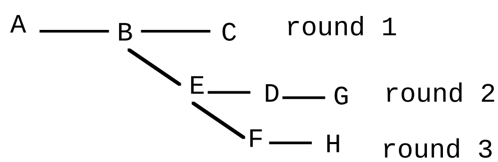


FIG 8

example 4

Fig 9 shows the depth-first search starting from E in Fig 4.
Fig 10 shows the depth-first search starting from G in Fig 4.

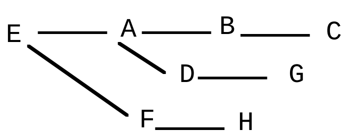


FIG 9

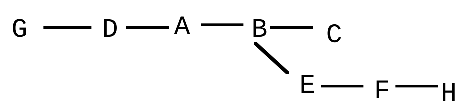


FIG 10

connection between the number of vertices and number of edges in a tree

A tree has one more vertex than edge, i.e.,

$$V = E + 1 \tag{1}$$

It follows that in a graph, if $V \neq E+1$ then the graph can't be a tree.
On the other hand, non-trees can have this property too so if a graph does have $V = E + 1$ the graph is not necessarily a tree. See problem 7(b).

why (1) holds

Look at the tree in Fig 11. Imagine growing it (Fig 12) with the depth-first spanning tree algorithm.

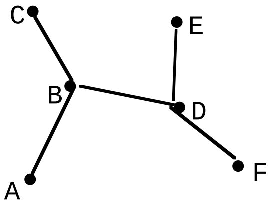
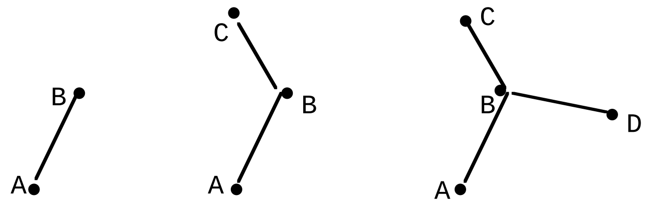


FIG 11



2 vertices 1 edge — STEP 1

3 vertices 2 edges — STEP 2

4 vertices 3 edges — STEP 3

etc

Growing the tree in Fig 11
FIG 12

You *start* (Fig 12, step 1) with one less edge then vertex, namely with V = 2, E = 1. As the tree grows, at each stage, you join a new vertex to the growing tree, i.e., you add one more edge and one more vertex. So you *continue* to have one less edge than vertex. So that's what you have when the tree is fully grown.

example 5

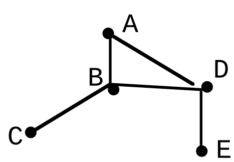
Suppose a graph has 12 vertices and 15 edges. Then it can't be a tree because $V \neq E+1$.

Suppose a graph has 12 vertices and 11 edges. It may or may not be a tree.

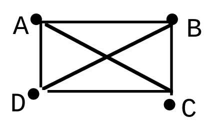
PROBLEMS FOR SECTION 2.3

1. By inspection (forget fancy algorithms) decide how many spanning trees each graph has.

(a)

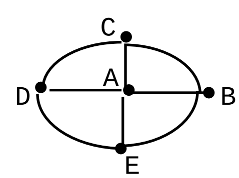


(b)

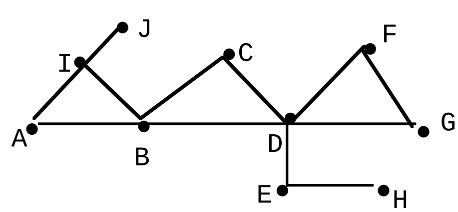


2. If a graph consists of a cycle with n edges how many spanning trees does it have.

3. Find a spanning tree for the graph in the diagram.
(a) breadth-first starting from A
(b) breadth-first starting from D
(c) depth-first starting from A
(d) depth-first starting from D



4. Find a spanning tree for the graph in the diagram.
(a) breadth-first from A
(b) breadth-first from D
(c) depth-first from A
(d) depth-first from D



5. Here's the adjacency matrix for a connected graph. Without actually drawing the graph, find a spanning tree starting from v1
(a) with depth-first (b) with breadth-first

	v1	v2	v3	v4	v5
v1	0	1	0	1	0
v2		0	1	1	1
v3			0	0	0
v4				0	0
v5					0

(matrix is symmetric)

6. Here are the adjacency matrices for two graphs. For each, without drawing a picture, use the spanning tree algorithm to decide if the graph is connected. If it isn't connected, how many components does it have?

(a)

	A	B	C	D	E	F
A	0	0	1	1	1	0
B	0	0	1	0	1	0
C	1	1	0	0	0	1
D	1	0	0	0	0	1
E	1	1	0	0	0	0
F	0	0	1	1	0	0

(b)

	A	B	C	D	E	F	G	H	I	J
A									1	
B				1						
C					1			1		
D		1								
E			1							1
F									1	
G									1	
H			1							1
I	1					1	1			
J					1			1		

7. We know that the following statement is true.
(*) If a graph is a tree then $V = E + 1$

(a) Find a counterexample to *dis*prove the following statement.
(**) If $V = E+1$ then the graph is a tree.

(b) True or False. And defend your answer.
(***) If $V \neq E + 1$ then the graph is *not* a tree.

8. True or False
(a) If G has 32 edges and 28 vertices then G is not a tree.
(b) If G is not a tree then G is not connected and has at least one cycle.

9. If G is connected with 13 vertices and 17 edges, how many vertices and edges are there in a spanning tree for G.

SECTION 2.4 PLANAR GRAPHS

definition of a planar graph

Fig 1 shows two edges crossing as opposed to Fig 2 where four edges have common endpoint T and no edges cross. A *planar graph* is one which can be drawn (*redrawn* if necessary) to *not* have crossing edges. The graph in Fig 2 is planar. The graph in Fig 1 is also planar because it can be redrawn (Fig 3) to avoid edges crossing. The graph in Fig 4 is planar because it can be redrawn so that no edges cross (Fig 4A).

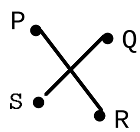


FIG 1

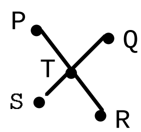


FIG 2

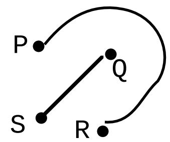


FIG 3

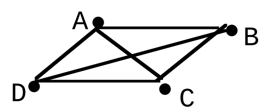


FIG 4

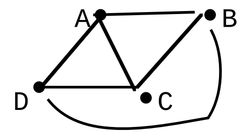


FIG 4A

showing that a graph is non-planar

I'll show that the graph in Fig 5 is not planar by showing (in a systematic fashion) that it can't be redrawn without edges crossing.

Begin by drawing the largest circuit possible; Fig 6 shows one which includes all the vertices. I still have to put in edges AE, BF, CD. Suppose AE is drawn *in*side (Fig 7A). Then BF must go *out*side to avoid crossing AE (Fig 7B). But now CD can't be drawn without either crossing AE inside or FB outside. So Fig 5 is nonplanar.

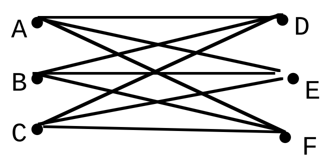


FIG 5

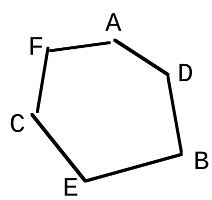


FIG 6

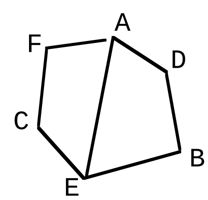


FIG 7A

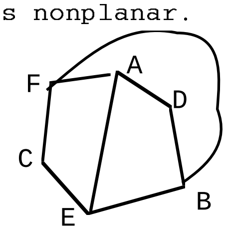


FIG 7B

warning

It is *not* convincing (especially on an exam) to try to show that a graph is non-planar by poking at its edges and simply declaring that they can't be redrawn without crossing. If you try to redraw, it must be with this systematic approach . An ad hoc approach may not be convincing to the reader.

faces of a planar graph

A planar graph divides the plane into disjoint regions called *faces*. One of them is the "outside" of the graph called the *unbounded* face or *exterior* face. The other faces are *interior* or *bounded* faces.

Fig 8 shows a planar graph with 4 faces; face IV is the unbounded face.
Fig 9 has 2 faces; II is the unbounded face.
If edge AD is added to Fig 9 as shown in Fig 10, the new graph still has 2 faces.

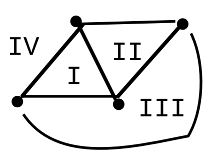


FIG 8

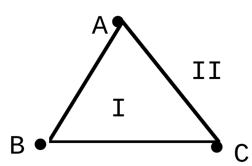


FIG 9

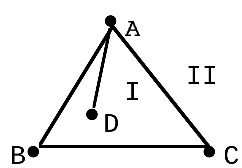


FIG 10

Euler's theorem

(1) $\boxed{\text{In any planar connected graph, } V - E + F = 2.}$

In other words, number of vertices − number of edges + number of faces = 2.

Note that the F in (1) *includes* the external face.
For the particular instance in Fig 11,
$V = 5$, $E = 7$, $F = 4$ (3 interior faces, 1 exterior face)
$V - E + F = 5 - 7 + 4 = 2$

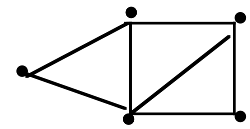


FIG 11

why (1) holds

Let G be a planar connected graph with V vertices (Fig 12 gives an example). Consider a spanning tree for G (Fig 13). The tree has V vertices, V − 1 edges and one face (the unbounded face). So (1) holds for the spanning tree since in the tree,

$$V - E + F = V - (V-1) + 1 = 2$$

Now add edges to the tree to get G back again. Each time you add an edge (Fig 14) the number of vertices stays the same, the number of edges goes up by 1 and the number of faces goes up by 1. So during the growth process, (1) continues to be true and so it holds at the end of the process for G itself.

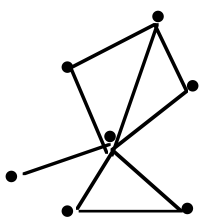


FIG 12

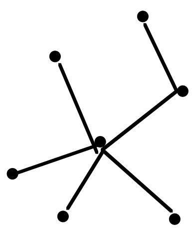


FIG 13

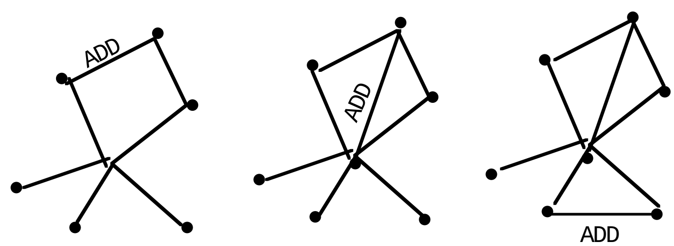
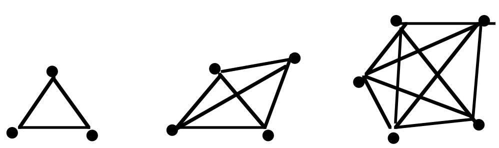


FIG 14 Going from tree to G

complete graphs

A graph which contains one edge between each pair of vertices is called *complete*. The complete graph with n vertices is called K_n (Fig 15).



K_3 K_4 K_5

Fig 15 Some complete graphs

bipartite and complete bipartite graphs

A graph consisting of two sets of vertices so that edges go only from one set to the other, with no edges within each set, is called *bipartite* (Fig 16).

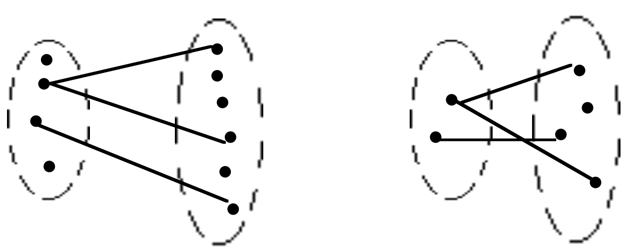
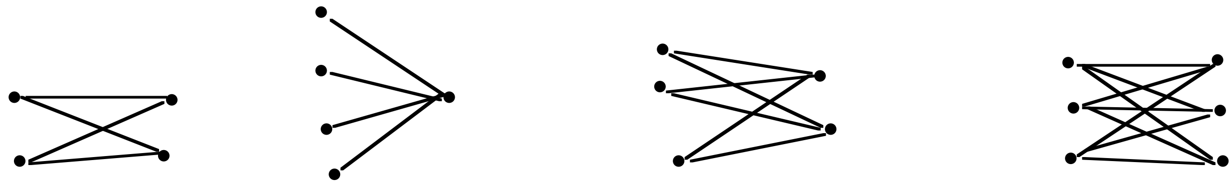


FIG 16 Two bipartite graphs

A graph is *complete bipartite* if every vertex in one set is joined by one edge to every vertex in the other set. In that case, if one set contains m vertices and the other set contains n vertices, the graph is called $K_{m,n}$ (same as $K_{n,m}$) (Fig 17).



$K_{2,2}$ $K_{4,1}$ (alias $K_{1,4}$) $K_{3,2}$ (alias $K_{2,3}$) $K_{3,3}$

FIG 17 Some complete bipartite graphs

the famous nonplanar graphs $K_{3,3}$, K_5 and beyond

The graphs K_5 and $K_{3,3}$ are nonplanar. Furthermore each is a dividing line between planar and nonplanar in the following sense:

K_2, K_3, K_4 are planar; K_5, K_6, ... are non-planar.

$K_{m,n}$ is planar if m < 3 or n < 3 and nonplanar if *both* are ≥ 3.

For example,

$K_{2,2}$, $K_{2,3}$, $K_{2,100}$ are planar.

$K_{3,3}$, $K_{3,4}$, $K_{4,4}$, $K_{4,100}$ are nonplanar.

why this holds

I already showed that $K_{3,3}$ is nonplanar (go back to Fig 5 and the paragraph above it). Similarly (see problem 8) it can be shown that K_5 is nonplanar.

The "earlier" versions are planar: Fig 18 shows $K_{m,2}$ redrawn without edges crossing. Fig 18A shows K_4 redrawn without edges crossing

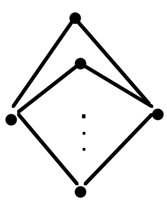


FIG 18 $K_{m,2}$ redrawn

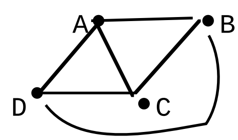


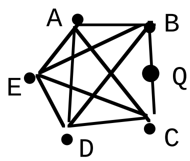
FIG 18A K_4 redrawn

The later complete bipartite graphs, like $K_{4,3}$, $K_{5,3}$, ..., and the later complete graphs, like K_6 , K_7, ..., are nonplanar because they contain $K_{3,3}$ and K_5 respectively as a subgraph and any graph which contains a nonplanar subgraph must itself be nonplanar.

graphs homeomorphic to K_5 and $K_{3,3}$

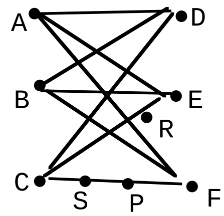
Look at the graph in Fig 19. It is not K_5 because the "extra" vertex Q was added which splits edge BC into BQ and QC. But Fig 19 is called *homeomorphic* to K_5. If Q is removed and edges BQ and QC merged into the single edge BC then it becomes K_5.

Similarly, Fig 20 is not $K_{3,3}$ but it is homeomorphic to $K_{3,3}$ because if the extra vertices P, S, R are removed and edges merged, it would become $K_{3,3}$.



homeomorphic to K_5

FIG 19



homeomorphic to $K_{3,3}$

FIG 20

More generally, graphs G_1 and G_2 are called homeomorphic if one of them can be turned into the other by removing one or more vertices *of degree 2* (like P, S, R in Fig 20)) and merging edges.

Any graph homeomorphic to K_5 or $K_{3,3}$ is also non-planar.

footnote Why does the definition of homeomorphic insist that the extra vertices be of *degree 2*?

If we allow higher-degree extras the new graph can become planar.For example, the graph in Fig 21 is K_5 with extra vertices of degree 4 (see P, Q, R, S, T). But the Fig 21 graph is planar and K_5 is not planar.

We don't want a planar graph to be homeomorphic to a non–planar graph.

This never happens if you stick to extra vertices of degree 2. So we make the definition accordingly.

The definition is designed to make the following theorem hold.

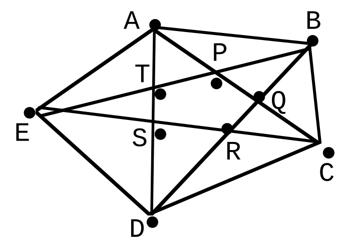


FIG 21

Kuratowski's theorem (characterizing nonplanar graphs)

> A graph is nonplanar if and only if it contains as a subgraph either K_5 or $K_{3,3}$ or something homeomorphic to one of them.

Part of the proof of Kuratowski's theorem is easy. Any graph which contains K_5 or $K_{3,3}$ or something homeomorphic to one of them is nonplanar because any graph which contains a nonplanar subgraph must itself be nonplanar. The other half of the proof is very long and is omitted.

warning K's theorem does *not* say that every non-planar graph contains $K_{3,3}$ or K_5. (After all, the graph in Fig 19 is non-planar and contains neither $K_{3,3}$ nor K_5.) It says that every non-planar graph contains $K_{3,3}$ or K_5 or a homeomorphic copy of one of them.

example 1

The graph in Fig 22 is not K_5 (because of the multiple edges between B and Q). But it contains K_5 as a subgraph so it's nonplanar.

Fig 23 is not K_5 (because of Z). It doesn't contain K_5 as a subgraph. But it is homeomorphic to K_5 so it's nonplanar.

Fig 24 is not K_5 and does not contain K_5 as a subgraph. But it does contain a subgraph (the one in Fig 23) which is homeomorphic to K_5 so it's nonplanar.

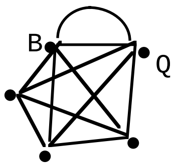


FIG 22
Contains K_5

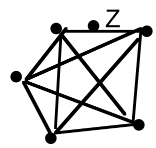


FIG 23
Homeomorphic to K_5

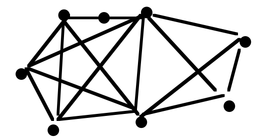


FIG 24
Contains a graph homeomorphic to K_5

example 2

Show that the graph in Fig 25 is nonplanar by finding a subgraph homeomorphic to $K_{3,3}$ or K_5.

The graph can't contain K_5 or something homeomorphic to K_5 since it doesn't have five vertices of degree 4 or more. So look for a buried $K_{3,3}$.

Vertex A is connected to B,F,E. So I'll look for two more vertices connected to B,F,E.

Vertex D can be connected to B via C, to F via I, and directly to E (Fig 26).

Vertex H can be connected to B via J and G, directly to F, and to E through J (Fig 26). But using J *twice* makes it an extra vertex whose degree is *more than 2* so Fig 26 is *not* homeomorphic to $K_{3,3}$.

The degree 2 requirement in the definition of homeomorphism means that in this search process *a vertex can't be used as an extra more than once*.

I tried re-routing H to B via C instead of J and G, leaving J for the H to E part. But then C is used twice. No good.

Then I gave up on H and tried J instead. Vertex J is connected to B via G, to F via H, and directly to E.

Fig 27 is homeomorphic to $K_{3,3}$ and it's a subgraph of Fig 25.

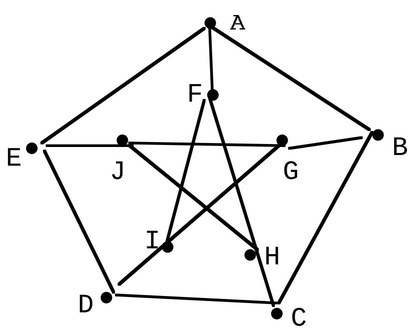


FIG 25

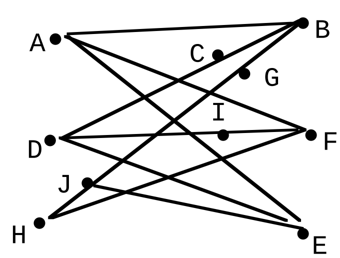


FIG 26
Not homeomorphic to $K_{3,3}$

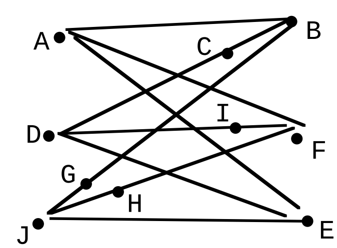


FIG 27
homeomorphic to $K_{3,3}$

Note that Fig 27 is a subgraph of Fig 25 and *with respect to the subgraph*, the extra vertices G, H, I, C have degree 2 so the subgraph is homeomorphic to $K_{3,3}$. It's irrelevant that *with respect to the original graph*, G, H, I, C have degree 3.

the dual graph

If G is a planar connected graph then G has a dual graph G^* constructed as follows. (1) In every face of G, including the unbounded face, place a vertex of G^*. If the face is named P then it's convenient to name the vertex P also.
(2) If edge e in G is on the boundary between faces P and Q, draw edge e^* in G^* joining vertices P and Q.

In other words, faces in G correspond to vertices in G^*, and vertices in G^* are joined by an edge if the corresponding faces in G have a common boundary.
It's easier to understand with an example.
Fig 28 shows a planar connected graph; Fig 29 is its dual.

original graph	*dual*
faces P, Q, R	vertices P, Q, R
edge 4 on boundary between faces P, Q	edge 4^* joining vertices P, Q
edge 1 on boundary between faces P, R	edge 1^* joining vertices P, R
edge 5 on boundary between faces Q, R	edge 5^* joining vertices Q, R
etc.	

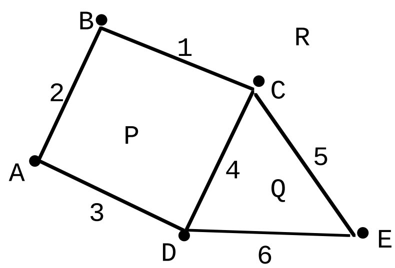


FIG 28 Graph G

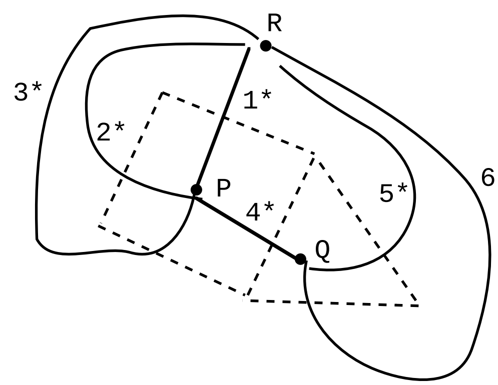


FIG 29 Graph G^*

warning

In Fig 28, edge 2 is a boundary between face P and the unbounded face R. Don't forget edge 2^* in the dual.
The dual should have an edge for each edge in the original.

Furthermore if you find the dual of G^* the result will be the dotted graph in Fig 29, namely G. So $G^{**} = G$.
In general, if G is a planar connected graph, then *G and G^* are duals of each other*, i.e., $G^{**} = G$.
Figs 30 and 31 show another graph and its dual. Edge 3 in Fig 30 is considered to be a boundary between face Q and itself and gives rise in G^* to an edge 3^* from vertex Q to itself, i.e., to a loop.

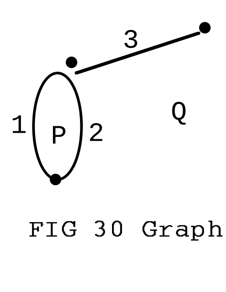


FIG 30 Graph G

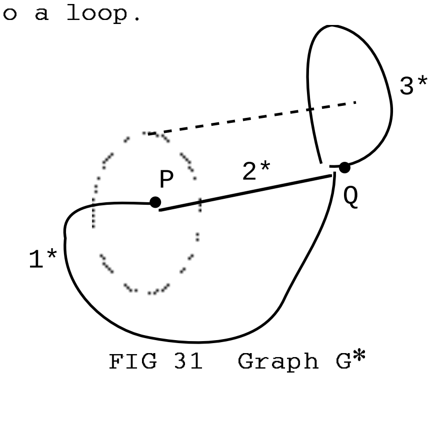
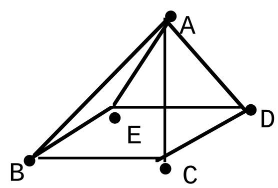


FIG 31 Graph G^*

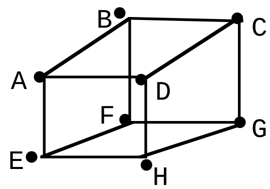
PROBLEMS FOR SECTION 2.4

1. Are these graphs planar or nonplanar?

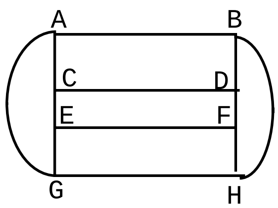
(a)



(b)

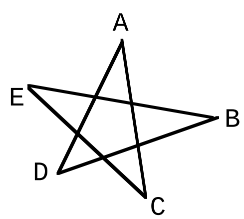


2. The diagram shows a planar graph where the exterior face is bounded by the cycle ABHGA. Redraw it, still with no edges crossing, so that the exterior face is bounded by ABDCA.

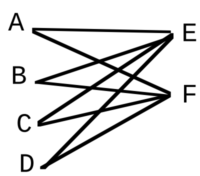


3. Redraw the graphs in the diagram to show that they're planar and then (since they're also connected) verify Euler's theorem .

(a)



(b)



4. True or False
   (a) Adding loops can never change a graph from planar to nonplanar.
   (b) Adding multiple edges can never change a graph from planar to nonplanar.
   (c) Removing loops can never change a graph from nonplanar to planar.
   (d) Removing multiple edges can never change a graph from nonplanar to planar.

5. Verify Euler's theorem for a tree with 1000 vertices.

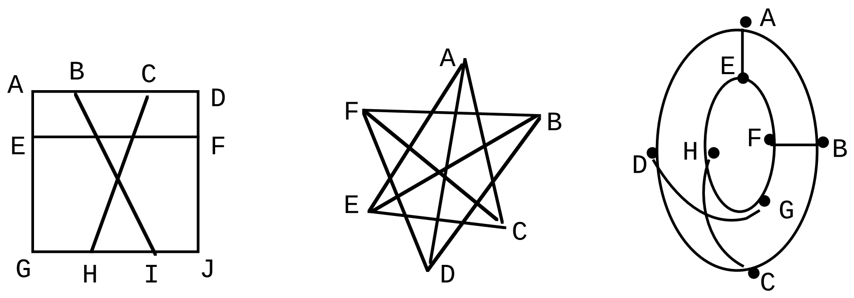
6. Suppose a planar connected graph has 9 vertices with degrees 2,2,2,3,3,3,4,4,5. How many edges? faces?

7. If a planar connected graph has 16 faces and all its vertices have degree 6 find the number of vertices.

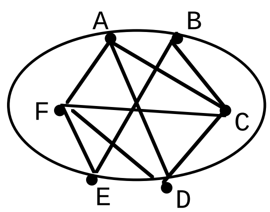
8. Show that K_5 is non-planar by systematically trying to re-draw it.

9. Try to redraw each of the following (systematically) to see if it's nonplanar. For each nonplanar one, find a subgraph that is $K_{3,3}$ or K_5 or homeomorphic to one of them.

(a) (b) (c)



10. Show that the graph in the diagram is nonplanar by finding a subgraph which is K_5 or homeomorphic to K_5.



11. Show that if any edge is removed from K_5 the result is a planar graph

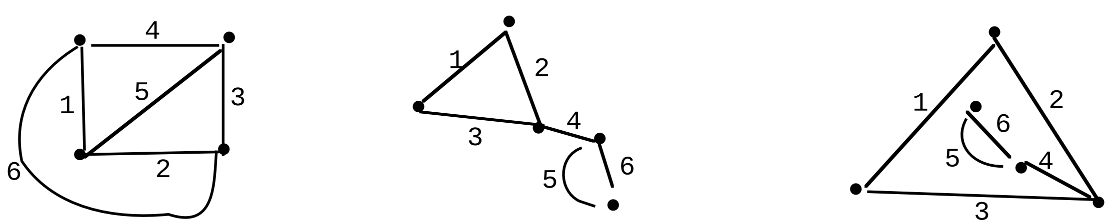
(a) using Kuratowski's theorem

(b) by actually redrawing the remainder with no edges crossing

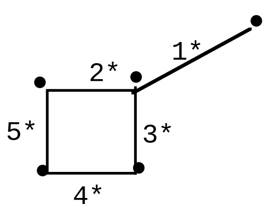
12. What happens if you remove an edge from K_6. Is the result planar or non-planar.

13. Draw the dual graph.

(a) (b) (c)

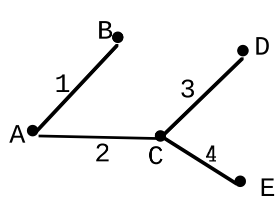


14. Given the dual in the diagram, draw the original.



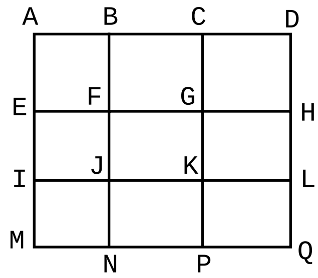
15. (a) Draw the dual of the tree in the diagram.

(b) Recopy the dual to get a fresh start and draw the dual of the dual to check you get the original back again.



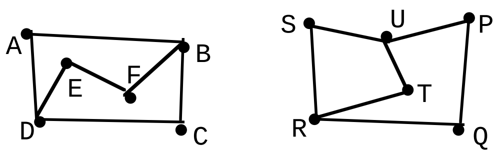
REVIEW PROBLEMS FOR CHAPTER 2

1. Find a spanning tree for the graph in the diagram breadth-first from F

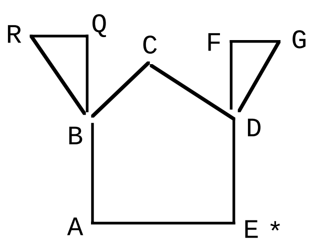


2. Suppose a spanning tree for a connected planar graph G consists of edges $e_1, e_2, \ldots, e_n$
   True or False and defend your answer.
   The edges $e_1^*, e_2^*, \ldots e_n^*$ form a spanning tree for the dual G^*.

3. Are these graphs isomorphic? Explain.



4. Find an Euler cycle if possible starting from vertex E. If not possible explain why not.



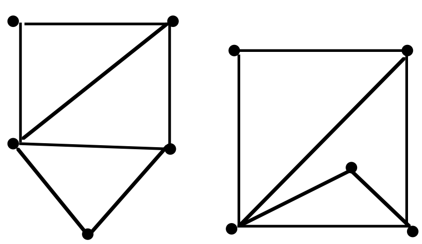
5. I have a connected graph G with 27 vertices and 50 edges.
   Fill in each blank with MUST, MIGHT, or CAN'T.

(a) If I throw away 25 edges, the resulting graph _______ be a spanning tree for G.

(b) If I throw away 24 edges, the resulting graph _______ be a spanning tree for G.

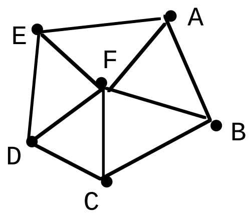
(c) If I throw away one vertex and 25 edges, the resulting graph __________ be a spanning tree for G.

6. Here are two isomorphic graphs.
   (a) Draw their duals
   (b) Show that their duals are not isomorphic.
   (c) Does that bother you? If the originals are really the "same" shouldn't they have the same duals.

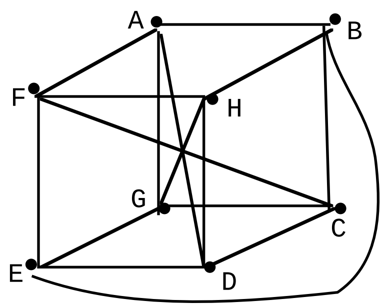


7. If a planar connected graph has 16 edges and each vertex has degree 4, how many faces does it have.

8. A planar connected graph G is called *self-dual* if G and G* are isomorphic. Is the graph in the diagram self-dual.



9. Show that the graph in the diagram (a cube plus 4 diagonals) is nonplanar by finding a subgraph which is $K_{3,3}$ or homeomorphic to $K_{3,3}$.



CHAPTER 3 SOME GRAPH ALGORITHMS

SECTION 3.1 WARSHALL'S ALGORITHM

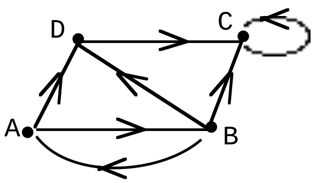
digraphs

A graph with one-way edges is called a *directed graph* or a *digraph*.

For the digraph in Fig 1, the adjacency matrix contains a 1 in row B, col C to indicate edge BC *from* B *to* C. There's a 0 in row C, col B since there is no edge CB *from* C *to* B.

In general, the adjacency matrix of a digraph is not necessarily symmetric.

An undirected graph can always be redrawn as a directed graph. Just replace each (two-way) edge by two one-way edges (Fig 2)



	A	B	C	D
A	0	1	0	1
B	1	0	1	1
C	0	0	1	0
D	0	0	1	0

FIG 1

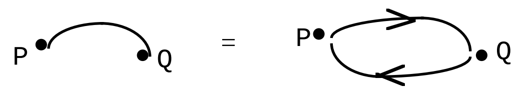


FIG 2

paths and cycles in a digraph

In the *underlying graph* of Fig 1 (where the edges are not directed) there is a path between A and C; each vertex is reachable from the other. But in the *digraph* (where the edges are directed) there is a path from A to C but not from C to A.

Similarly, in the underlying graph of Fig 1 there's a cycle ABDA. But in the digraph there is no cycle ABDA since there is no edge directed from D to A.

the reachability matrix M_∞

If M is the adjacency matrix of a digraph then an entry of 1 in row i, col j indicates an edge $v_i v_j$, i.e., a path from v_i to v_j with just one edge. In this section I'll extract from M a new matrix called the *reachability matrix*, denoted M_∞, in which an entry of 1 in row i, col j indicates a path (with one or more edges) from v_i to v_j, and an entry of 0 means no path at all. In other words, the reachability matrix indicates whether you can get from here to there. (Some books call M_∞ the *transitive closure* of M.)

footnote

It may seem odd but frequently you can't get from here to *here*. In Fig 1 there's a path from A to A, namely ABA, and there's a path from C to C, a loop, but there is no path from D to D.

Maybe it's not so odd after all. If the vertices in a graph represent people and an edge from A to B means that B is the parent of A then there can't be a path like A P Q A, from A to itself, since that would make A her own great grandmother.

Boolean arithmetic

To compute the matrix M_∞ from the adjacency matrix you'll be dealing entirely with 0's and 1's and it will turn out that Boolean arithmetic will be pertinent. It differs from ordinary arithmetic only in that

$$1 + 1 = 1.$$

In fact the new law 1 + 1 = 1 together with the old law 1 + 0 = 1 means that

$$1 + \text{anything} = 1$$

All sums in this section are intended to be Boolean.

Warshall's algorithm for finding the reachability matrix M_∞ for a digraph

Start with a digraph with n vertices.
Here's the idea of the algorithm.
Begin with the adjacency matrix M which indicates which pairs of vertices are directly connected.
The first round will get a matrix M_1 which indicates which pairs of vertices are connected by a path using v_1 as the only possible intermediate point (i.e., connected by an edge from here to there or by a path from here to v_1 to there).
The next round gets matrix M_2 which indicates which pairs of vertices are connected by a path allowing only v_1 and v_2 as possible intermediate points etc.

(1)

> In general here is the *loop invariant* (something that's true after every round):
> At round k, you will get a matrix M_k. Look at the entry say in row 3, col 5.
> An entry of 1 means that there is a path from v_3 to v_5 with $v_1, \ldots, v_k$ as the only possible intermediates. An entry of 1 means that there is no path from v_3 to v_5 that uses only if only $v_1, \ldots, v_k$ as intermediates,

Here's the algorithm for finding M_∞ starting with a digraph with n vertices and adjacency matrix M.

> Set $M_0 = M$.
> To get M_1, for every row in M_0 that has a 1 in col 1, add (Boolean) row 1 to that row; i.e., look down col 1 and if there's a 1 in a row, add row 1 to that row.
>
> To get M_2, for every row in M_1 with a 1 in col 2, add row 2 to that row.
>
> In general, to go from M_{k-1} to M_k, for every row in M_{k-1} that has a 1 in col k, add row k to that row.
> Continue until you have M_n.

At the end of the algorithm,

(2) $\boxed{M_\infty = M_n}$

I'll do an example first and then show why the algorithm works, i.e., show why (1) and (2) hold.

example 1

Start with a digraph with this adjacency matrix M.

$$M = \begin{matrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

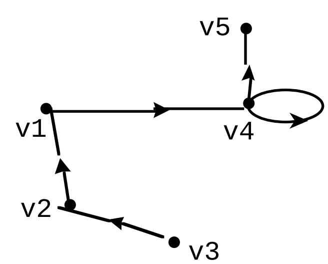


FIG 3

That's M_0.

round 1 To go from M_0 to M_1.
Look at col 1 in M_0. The only row with a 1 in col 1 is row 2. So add row 1 to row 2

and leave the other rows alone. Note that the Boolean addition amounts to replacing 0's in row 2 by 1's if there is a corresponding 1 in row 1.

$$M_1 = \begin{matrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

round 2 To go from M_1 to M_2.

Look at col 2 in M_1. The only row with a 1 in col 2 is row 3. So add row 2 to row 3 and leave the other rows alone.

$$M_2 = \begin{matrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

round 3 To go from M_2 to M_3.

Look at col 3 of M_2. There are no 1's in col 3 so leave M_2 alone, i.e., $M_3 = M_2$.

round 4 Look at col 4 of M_3. The first four rows contain 1's in col 4 so add row 4 to each of the first four rows (note that adding a row to itself can't change it).

$$M_4 = \begin{matrix} 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

round 5 Look at col 5 in M_4. The first four rows contain 1's in col 5 so add row 5 to them which doesn't change anything: $M_5 = M_4$. So

$$M_\infty = M_5 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

For instance, column 3 is all 0's so there are no paths to v_3 from anywhere; row 5 is all 0's so there are no paths from v_5 to anywhere.

why (1) holds

I'll use the graph from example 1 to illustrate the idea.

Here's why (1) holds for k = 1, i.e., for M_1.

As you go from M_0 to M_1 there are two ways in which entries of 1 appear in M_1.

(I) All the 1's in M_0 are still there in M_1

For example, the 1 in row 3, col 2 is a holdover. In M_0 (and therefore in M_1) it indicates an edge from v_3 to v_2, which you can call a path from v_3 to v_2 using no intermediates.

(II) Some 0's in M_0 might become 1's in M_1

For example, the 0 in row 2, col 4 of M_0 changed to a 1 because there was a 1 in row 2, col 1 (this told you to add row 1 to row 2) and there was a 1 in row 1, col 4 (which got added to the 0 entry and changed it to a 1) (Fig 4)

edge v2 to v1

edge v1 to v4

changes to a 1 in M_1

$$M_0 = \begin{matrix} 0 & 0 & 0 & \boxed{1} & 0 \\ \boxed{1} & 0 & 0 & \boxed{0} & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

FIG 4 See a 0 in M_0 change to a 1 in M_1

The 1 in row 2, col 1 (in M_0) indicates that there is an edge from v_2 to v_1.
The 1 in row 1, col 4 indicates that there is an edge from v_1 to v_4.
Put those two edges together and you know there is a path from v_2 to v_4 using v_1 as an intermediate, justifying the 1 in row 2, col 4 of M_1.

All in all, a 1 in row i, col j in M_1 signals a path from v_i to v_j either using no intermediate (if it's type (I)) or using v_1 as the only intermediate (if it's type (II)). And every such path is signaled like that in M_1.

Here's why (1) holds for M_2.

As you go from M_1 to M_2 there are two ways in which entries of 1 appear in M_2.

(I) All the 1's in M_1 are still there in M_2.

For example the 1 in row 2, col 4 is a holdover.

In M_1, and now in M_2, it indicates a path from v_2 to v_4 using v_1 as a possible intermediate.

(II) Some 0's in M_1 might become 1's in M_2.

For example, the 0 in row 3, col 4 of M_0 changed to a 1 because there was a 1 in row 3, col 2 (this told you to add row 2 to row 3) and there was a 1 in row 2, col 4 (which got added to the 0 entry and changed it to a 1) (Fig 5)

path v3 to v2
with v1 as the
only possible
intermediary

path v2 to v4

changes to a 1 in M 2

$$M_1 = \begin{matrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & \boxed{1} & 0 \\ 0 & \boxed{1} & 0 & \boxed{0} & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

FIG 5 See a 0 in M_1 change to a 1 in M_2

As I just proved, the 1 in row 3, col 2 (in M_1) indicates that there is a path from v_3 to v_2 with v_1 as the only possible intermediate.

And the 1 in row 2, col 4 indicates that there is a path from v_2 to v_4 with v_1 as the only possible intermediate.

Put the two paths together and you know there is a path from v_3 to v_4 using v_2 as an intermediate and maybe v_1 but no other intermediates.

All in all, a 1 in row i, col j in M_2 (whether type(I) or type (II)) signals a path from v_i to v_j with v_1 and v_2 as the only possible intermediates. And every such path is signaled like that in M_2.

So far, I have (1) true for M_1 and M_2. Similarly it's true for M_k in general.

why (2) holds

By (1), an entry of 1 in row i, col j in M_n indicates a path from v_i to v_j using $v_1, \ldots, v_n$ as the only possible intermediates. But the graph only has the n vertices $v_1, \ldots, v_n$. So this is your last chance for paths. So $M_n = M_\infty$.

mathematical catechism (you should know the answer to these questions)

question 1 After round 4 of Warshall's algorithm, what does an entry of 1 in row 3, col 6 in M_4 mean.

answer There is a path from v3 to v6 using only vertices v1, v2, v3, v4 as possible intermediates.

question 2 After round 4 of Warshall's algorithm, what does an entry of 0 in row 3, col 6 in M_4 mean.

answer There is no path from v3 to v6 using only vertices v1, v2, v3, v4 as possible intermediates. (There may be a path to be discovered in later rounds using more than v1, v2, v3, v4 as possible intermediates.)

warning Saying "no path from v3 to v6 *so far*" is not good enough. Too vague.

PROBLEMS FOR SECTION 3.1

1. Let M be the adjacency matrix of a digraph. Find as many entries of M_∞ as possible if
   (a) vertex v_2 is an isolated vertex
   (b) the digraph contains the (directed) cycle $v_1v_7v_8v_1$

2. If M_∞ contains 1's in row 1, col 2 and in row 2, col 5 what other 1's must it have.

3. Suppose a digraph has 9 vertices. And there is a path $v_7v_1v_5v_2v_3$ (there may be other paths but this is all you know about).
What entries can you determine in $M_0, \ldots, M_9$ from this information.

4. (a) Suppose there is a 1 in row 1, col 6 in M_3. What does that signify (quote the loop invariant).
   (b) Suppose there is a 0 in row 6, col 1 in M_3. What does that signify.

5. Find M_∞ if M is

(a) $\begin{matrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{matrix}$ (b) $\begin{matrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{matrix}$

6. Find the reachability matrix R_∞ if the adjacency matrix R is

(a) $\begin{matrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{matrix}$ (b) $\begin{matrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{matrix}$ (c) $\begin{matrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{matrix}$

7. Let

$$R = \begin{matrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

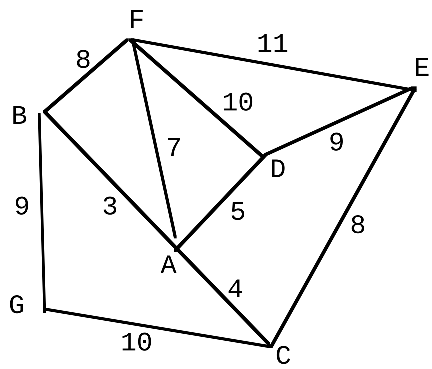
be the adjacency matrix of a digraph. Find R_∞ and then use it to decide if there's a path from (a) v_2 to v_4 (b) v_4 to v_2

SECTION 3.2 PRIM'S ALGORITHM

weighted graphs and weight matrices

Start with a graph with no loops and no multiple edges. Suppose each edge has a non-negative number associated with it called the edge *weight* or *cost.* (The weight of edge AB might be the cost of building a telephone line between A and B or the distance between A and B etc.)

Fig 1 shows a weighted graph with the corresponding *weight matrix*. The entry 4 in row A, col C (and row C, col A) is the weight of the edge AC. The entries on the diagonal are chosen to be 0, i.e., choose the "distance" from a vertex to itself to be 0. If two different vertices are not connected by an edge, put ∞ in the weight matrix, indicating no edge at any cost.



	A	B	C	D	E	F	G
A	0	3	4	5	∞	7	∞
B	3	0	∞	∞	∞	8	9
C	4	∞	0	∞	8	∞	10
D	5	∞	∞	0	9	10	∞
E	∞	∞	8	9	0	11	∞
F	7	8	∞	10	11	0	∞
G	∞	9	10	∞	∞	∞	0

FIG 1

minimal-weight spanning trees

A connected weighted graph has many spanning trees (each representing say a system of telephone lines or a system of roads allowing cities to be linked without wasting money on cyclic links). Each spanning tree has a weight, the sum of all the edge weights (the total cost of the telephone system, total miles of roadway). A *minimal spanning tree* for a connected weighted graph is a spanning tree with minimum weight (links the cities with the cheapest phone system, with the least asphalt).

Prim's (greedy) algorithm for a minimal spanning tree in a weighted connected graph

Given a weighted connected graph.
Here's the idea of the algorithm.

> Pick any initial vertex and start to grow a tree.
> Add vertices and edges one at a time as follows.
> At each step find all vertices not yet in the growing tree but connected by edges to some vertex already in the tree. Pick the one linked by the cheapest edge (i.e., be greedy) and grow the tree by adding that vertex and edge.

example 1

For a small graph you can run Prim's algorithm by inspection. Look at the graph in Fig 1 again. I'll find a minimal spanning tree starting with vertex G.

Vertices B and C are connected to G by edges. The cheaper connection is with B. So add vertex B and edge BG (Fig 2)

The next candidates are vertex C and edge CG, vertex F and edge BF, vertex A and edge AB. Cheapest is A and edge AB (Fig 3).

Now choose either vertex F and edge FB or vertex F and edge FA or vertex D and edge DA or vertex C and edge CA or vertex C and edge CG. Cheapest is vertex C and edge AC (Fig 4);

Then add vertex D and edge AD (Fig 5); vertex F and edge AF (Fig 6); vertex E and edge EC (Fig 7).

The total cost of the minimal spanning tree in Fig 7 is 36 (add the weights).

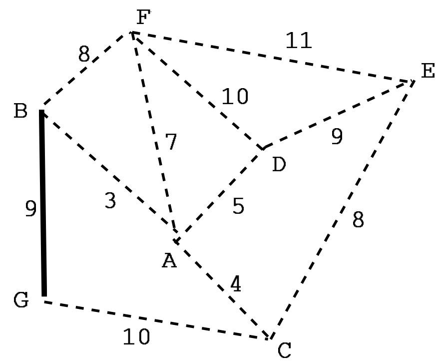


FIG 2

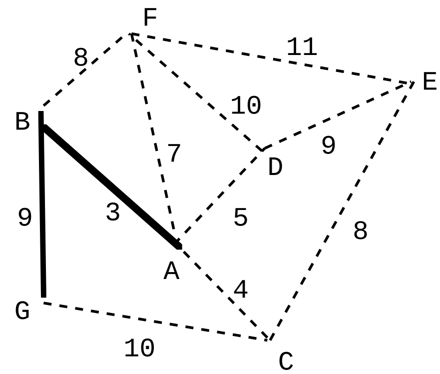


FIG 3

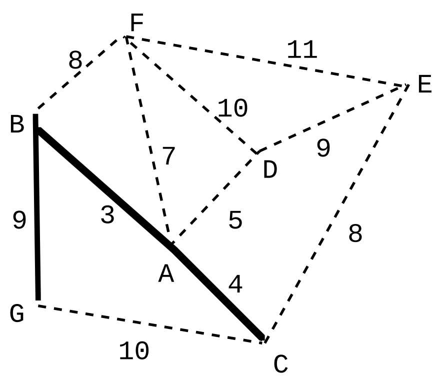


FIG 4

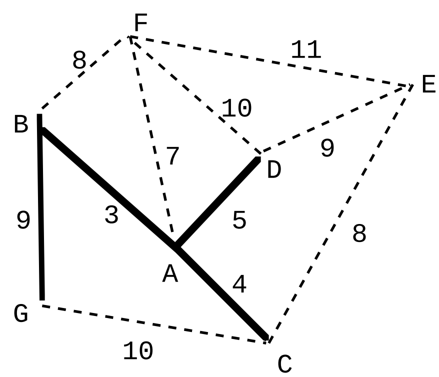


FIG 5

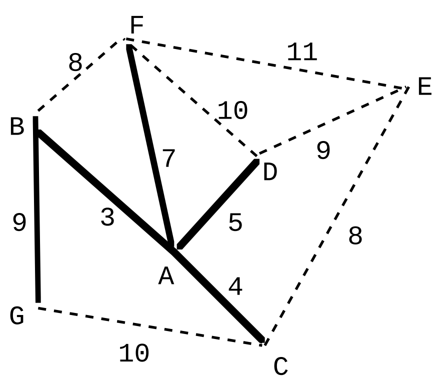


FIG 6

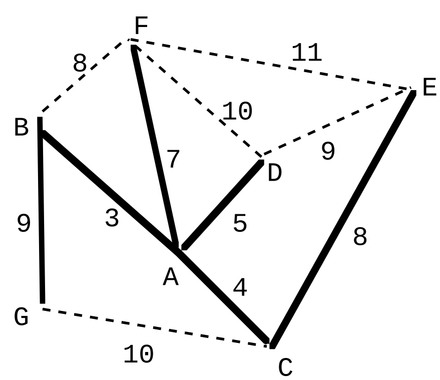


FIG 7

Prim programmed (how to do it in a large graph)

I'll repeat example 1, again starting with vertex G. Here's the weight matrix repeated.

	A	B	C	D	E	F	G
A	0	3	4	5	∞	7	∞
B	3	0	∞	∞	∞	8	9
C	4	∞	0	∞	8	∞	10
D	5	∞	∞	0	9	10	∞
E	∞	∞	8	9	0	11	∞
F	7	8	∞	10	11	0	∞
G	∞	9	10	∞	∞	∞	0

And at every round, for each vertex (except the one I started from) I'll keep track of these *loop invariants*.

status	Tells you if the vertex been picked for the tree yet I've typed √ for picked and x for not picked.
link	For any *unpicked* vertex v, link v is the vertex in the growing tree joined to v by the cheapest edge. In other words, if unpicked vertex v is going to join the growing tree *now*, the cheapest way to go would be to add the edge from v to link v.
cost	For any *unpicked* vertex v, cost v is the weight of that cheapest connection. It might improve as more vertices are picked.

round 0

Here's the initial table.
I put all the vertices except G in the table.
Each vertex has G has its link because G is the only picked vertex so far.
The costs come from col G (or row G) in the weight matrix

	A	B	C	D	E	F
status	x	x	x	x	x	x
link	G	G	G	G	G	G
cost	∞	9	10	∞	∞	∞

round 1

Of all the unpicked vertices, B is the one with the smallest cost (i.e., would be the cheapest to add right now). Pick B. And update the table like this.

Update B
Change its status to picked.

Update A
At the last round, A had link G with cost ∞. This means that at that stage if we added A to the tree by adding edge AG, the cost would be ∞ and that's the best A could have done.
But now B is available to be a link. Let's see if A can do better by linking to B.
cost A = ∞
AB = 3, better than ∞.
So change link A from G to B and change cost A from ∞ to 3.

Update C:
cost C = 10
CB = ∞, not an improvement.
Leave link C and cost C unchanged.

Update D:
DB = ∞, can't be an improvement
Leave link D and cost D unchanged.

Update E
EB = ∞, can't be an improvement.
No change

Update F
cost F = ∞
FB = 8, an improvement
Set link F = B, cost F = 8

	A	B	C	D	E	F
status	x	√	x	x	x	x
link	B	G	G	G	G	B
cost	3	9	10	∞	∞	8

round 2

Of all the unpicked vertices, A is the one with the smallest cost. Pick A. And update.

Update A
Change its status to picked.

Update C
At the last round, C had link G with cost ∞. This means that at that stage if we added C to the tree by adding edge CG, the cost would be ∞ and that's the best C could have done.

But now A is available. Can C do better by linking to A.
CA = 4, an improvement.
Set link C = A, cost C = 4

Update D
Cost D = ∞
DA = 5, better
Set link D = A, cost D = 5

Update E
EA = ∞, can't be better
No change

Update F
cost F = 8
FA = 7, better
Set link F = A, cost B = 7

	A	B	C	D	E	F
status	√	√	x	x	x	x
link	B	G	A	A	G	A
cost	3	9	4	5	∞	7

round 3

Pick C (because it has the smallest cost of all the unpicked vertices))

Update D
DC = ∞, DC will not be < cost C. No change.

Update E
EC = 8, cost E = ∞, EC < cost E
Set link E = C, cost E = 8

Update F
FC = ∞, FC will not be < cost F. No change.

	A	B	C	D	E	F
status	√	√	√	x	x	x
link	B	G	A	A	C	A
cost	3	9	4	5	8	7

round 4

Pick D

Update E
ED = 9, cost E = 8, ED is not < cost E, no change

Update F
FD = 10, cost F = 7, FD is not < cost F, no change

	A	B	C	D	E	F
status	√	√	√	√	x	x
link	B	G	A	A	C	A
cost	3	9	4	5	8	7

round 5

Pick F.
Update E
EF = 11, cost E = 8, no change

	A	B	C	D	E	F
status	√	√	√	√	x	√
link	B	G	A	A	C	A
cost	3	9	4	5	8	7

round 6

Pick E and stop. All the vertices have now been picked.

	A	B	C	D	E	F
status	√	√	√	√	√	√
link	B	G	A	A	C	A
cost	3	9	4	5	8	7

Join each vertex to its final link to get edges AB, BG, CA, DA, EC, FA. They are the edges in a minimal spanning tree (Fig 7). The weight of the tree is the sum of the final costs, namely 3 + 9 + 4 + 5 + 8 + 7 = 36

Here's a summary of the algorithm.

To start, pick an initial vertex. Say it's v_0.

For the other vertices v set
status v = x   (not picked yet)
link v = v_0
cost v = edge weight vv_0

At each round, among the unpicked vertices find the one with the smallest cost. Say it's v*. Change its status to √ (picked).
For each unpicked vertex v update like this.

If vv* < cost(v) then set link v = v* and cost v = vv*.

Repeat until all vertices are picked.

At the end of the algorithm there are the following conclusions.

The set of edges joining each vertex to its link is a minimal spanning tree and the sum of the cost v 's is its total weight

It can be shown that if the edge weights are all different then there is only one minimal spanning tree. If two edges have the same weight then there might be more than one minimal spanning tree but all minimal spanning trees have the same *weight*.

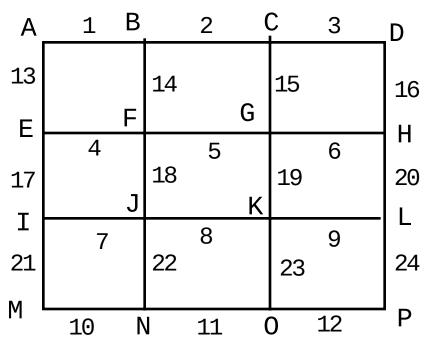
why the algorithm works

Prim's algorithm produces a spanning tree because it has the same form as the plain spanning tree algorithm from Section 2.3. But it takes a lot of proof to show that the tree is minimal just because you were greedy at each step. I'm leaving that out.

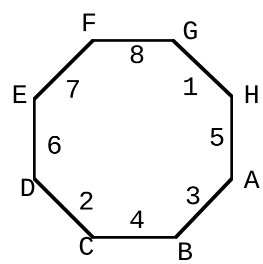
PROBLEMS FOR SECTION 3.2

1. Use Prim's algorithm to find a min spanning tree (but without formally keeping track of status, link, cost). For the initial vertex pick the alphabetically or numerically first vertex. List the edges in order as you find them and draw the tree.

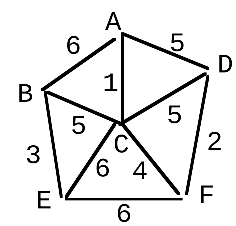
(a)



(b)



(c)



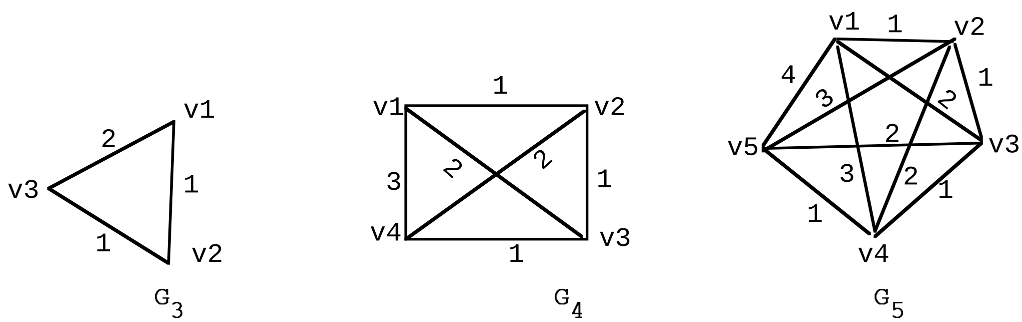
2. (a) Find the weight matrix for the graph in problem 1(c).
(b) Find a min spanning tree again, starting from vertex A, this time keeping track of status, link, cost at every step.

3. For the graph with the weight matrix below, find a min spanning tree, starting from vertex A, and its weight. Record status, link, cost at every step.

	A	B	C	D	E
A	0	2	∞	∞	3
B	2	0	1	∞	∞
C	∞	1	0	4	∞
D	∞	∞	4	0	5
E	3	∞	∞	5	0

4. Make up a graph that has more than one minimal spanning tree.

5. Start with the complete graph K_n and name the vertices successively $v_1, \ldots, v_n$. Suppose the cost of the edge $v_i v_j$ is $|i - j|$ (for example, the weight of $v_7 v_{11}$ is 4). Call the weighted graph G_n. The diagram shows the graphs G_3, G_4, G_5.



(a) Find a minimal spanning tree for G_5 and then in general for G_n. And find the weight of the min spanning tree.
(b) Repeat part (a) if the cost of the edge $v_i v_j$ is $i + j$ instead of $|i-j|$

6. (a) Look at the graph in problem 1(a). Of all the spanning trees which include the edge KO, find the one with minimum weight.
(b) Is the tree you just got in part (a) a minimal spanning tree for the graph.

SECTION 3.3 DIJKSTRA'S ALGORITHM

Dijkstra's algorithm for shortest distances and shortest paths from a given vertex

Start with a weighted conected graph.
Suppose you want to find a shortest path from vertex A to every other vertex.
Here's the idea of the algorithm.

Dijkstra's algorithm will grow a tree of shortest paths.
Start the tree with A. Add vertices one at a time as follows.
At each step look at all vertices not yet in the growing tree but connected by an edge to some vertex already in the tree. Pick the one that is closest to A and add it and its connecting edge to the tree.

example 1

For a small graph you can run Dijkstra's algorithm by inspection. Look at the graph in Fig 1. I'll find a tree of shortest paths from A to every other vertex.

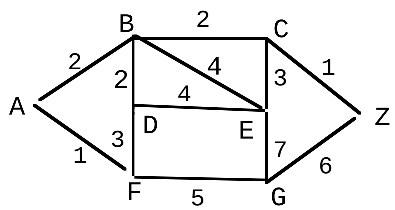


FIG 1

Start the tree with A.

Of all the other vertices, F is closest to A, via edge AF.
Add F and edge AF to the tree (Fig 2).

Consider vertices which can link to A or to F, namely B, D, G. Find their distances to A, namely AB = 2, AFD = 4, AFG = 6. Vertex B is closest to A.

Add B and edge AB to the tree (Fig 3).

Of all vertices which can link to A, B, or F (namely C, D, E, F) there's a tie between C (see path ABC) and D (see paths AFD, ABD) for which is closest to A. So I could add C and edge BC or I could add D and edge FD or BD. Toss a coin.

I'll add C and edge BC (Fig 4)

Of all vertices which can link to the growing tree, the one closest to A is D (see paths ABD or AFD, a tie).

I'll add D and edge FD (Fig 5).

Of all vertices which can link to the growing tree, the one closest to A is Z (see path ABCZ).

Pick Z. Add edge CZ (Fig 6).
(If the problem asked only for the shortest path from A to Z you would stop here.)

There's a tie between E and G.

Pick E (see path ABE). Add edge BE (Fig 7).
Pick G (see path AFG). Add edge FG (Fig 8.)

All vertices are picked. The end! You can read shortest paths from A to any other vertex from the tree in Fig 8.

If you want the shortest path from D to E, this tree doesn't help.

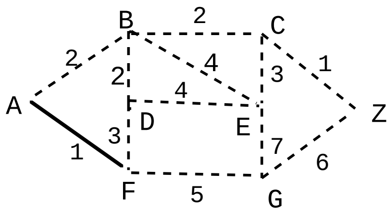


FIG 2

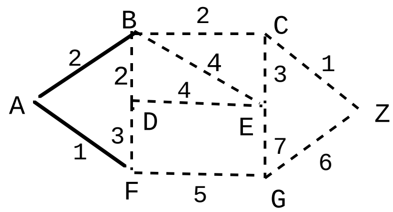


FIG 3

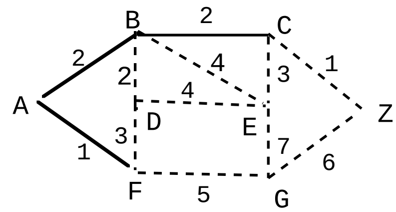


FIG 4

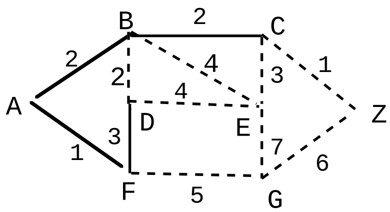


FIG 5

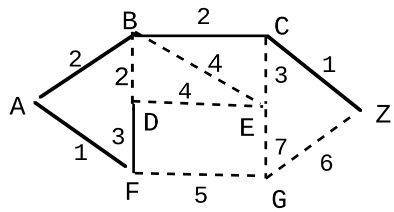


FIG 6

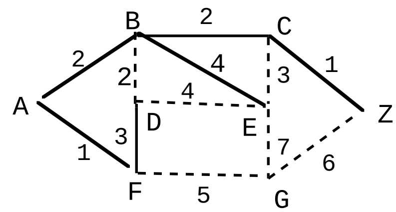
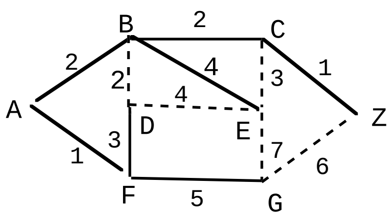


FIG 7



Dijkstra's tree of shortest paths from A
FIG 8

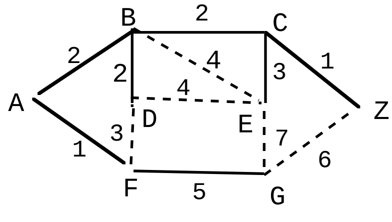
Dijkstra's algorithm vs. Prim's algorithm

The two algorithms both grow a spanning tree in a similar fashion, adding one vertex and edge at a time.

One difference is that Prim's algorithm can start with any initial vertex while Dijkstra's algorithm for shortest paths *from Q* must start the tree with initial vertex Q. The second difference is that Prim's algorithm grows the tree by picking the vertex linked by the cheapest edge while Dijkstra's algorithm for shortest paths *from Q* picks the vertex closest to Q.

To see the difference, run Prim's algorithm in Fig 1, beginning with initial vertex A. As the minimal spanning tree grows, you get Figs 1,2,3,4 as in Dijkstra's algorithm for shortest distances from A. But Prim's algorithm now chooses vertex Z and edge ZC because ZC is the cheapest edge which can be added on while Dijkstra's algorithm picks D and edge DF (or edge DB) because D is closest to A (i.e., because path AFD is the cheapest way to continue growing paths from A).

Fig 9 shows Prim's minimal spanning tree. It's of interest to the township which wants to link its towns with minimal asphalt. On the other hand, Dijkstra's tree of shortest paths from A in Fig 8 is of interest to people in town A who commute to surrounding towns.



Prim's minimal spanning tree
FIG 9

Dijkstra's algorithm programmed

Look at a new graph whose weight matrix is in Fig 10.

	A	B	C	D	E	F	G
A							
B	3						
C	6	2					
D	∞	4	1				
E	∞	∞	4	2			
F	∞	∞	2	∞	2		
G	∞	∞	∞	4	1	4	

FIG 10

I'll find shortest paths from A to every other vertex.
For each vertex other than A, keep track of status (picked or not picked yet), distance and link.

During the algorithm there are the following *loop invariants*.

(1) At the end of each round, for any vertex V, dist V is the shortest distance between A and V allowing only *picked* vertices as possible intermediates, and link V is the last vertex (just before V itself) on such a shortest path, i.e., it's where V attached itself to the tree.

(2) At the end of each round, if V is a *picked* vertex then dist V is *the* length of the shortest path between A and V (can't get a shorter path later in the algorithm as more vertices get picked).

round 0 Here's the initial table
I put all the vertices except A in the table.
Each vertex has A as its link because A is the only picked vertex so far.
The dists are just the weights of the edges to A.

	B	C	D	E	F	G
status	x	x	x	x	x	x
link	A	A	A	A	A	A
dist	3	6	∞	∞	∞	∞

round 1 The unpicked vertex with the smallest dist value is B so pick B. And update the table like this.

Update B
Change its status to picked.

Update C
At the last round, C had link A with dist 6. This means that at that stage if we added C to the tree by adding edge CA, the dist from A to C would be 6 and that's the best C could have done.
But now B is available to be a link.
Dist B = 3 so there is a path from A to B with length 3 (the best A to B path) and then B to C would be 2 more. That's a total distance of 5. So C can get a better path to A by linking to B.
Set link C = B
Set dist C = 5

warning
dist C is not 2, the single edge weight BC. Rather, dist C = dist B + BC = 3 + 2 = 5, the distance on a shortest-so-far path between A and C.

Update D
dist D = ∞ (i.e., there was no good path from A to D)
dist B + BD = 3 + 4 = 7
Set link D = B
Set dist D = 7.

Update E
dist E = ∞ (i.e. there was no good path from A to E).
But BE = ∞ also. So there can't be any improvement in a path from A to B to E.
No change in link E or dist E

Update F
BF = ∞. No change in link F or dist F

Update G
BG = ∞. No change in link G or dist G.

	B	C	D	E	F	G
status	√	x	x	x	x	x
link	A	B	B	A	A	A
dist	3	5	7	∞	∞	∞

Here's a summary.

Initially, pick A and for the other vertices V set

status V = x (not picked yet)
link V = A
dist V = weight of edge AV.

Among the unpicked vertices, pick the one with the smallest dist value. Call it V*
and change its status to picked.

For each remaining unpicked vertex V, update as follows:

(3) If dist V* + V*V < dist V
then set link V = V* and set dist V = dist V* + VV*

(The test in (3) is designed to see if the distance from the last pick, A, to an unpicked vertex V can be improved using the latest pick, V*, as a last intermediate. If so, switch to an improved dist for V and to V* as its new link.)

Repeat until all the vertices are picked.

round 2 Of all the unpicked vertices, C has the smallest dist value so pick C and update.
dist C + CD = 5 + 1 = 6, dist D = 7. Set link D = C, dist D = 6
dist C + CE = 5 + 4 = 9, dist E = ∞. Set link E = C, dist E = 9
dist C + CF = 5 + 2 = 7, dist F = ∞. Set link F = C, dist F = 7
dist C + CG = 5 + ∞ = ∞ No change

	B	C	D	E	F	G
status	√	√	x	x	x	x
link	A	B	C	C	C	A
dist	3	5	6	9	7	∞

round 3 Pick D and update

If you're interested only in the shortest distance and shortest path between A and D then you can stop here; as soon as a vertex is picked, its dist and link give final results.

dist D + DE = 6 + 2 = 8, dist E = 9 Set link E = D, dist E = 8
dist D + DF = 6 + ∞ = ∞ No Change
dist D + DG = 6 + 4 = 10, dist G = ∞ Set link G = D, dist G = 10

	B	C	D	E	F	G
status	√	√	√	x	x	x
link	A	B	C	D	C	D
dist	3	5	6	8	7	10

round 4 Pick F and update
dist F + FE = 7 + 2 = 9, dist E = 8 No Change
dist F + FG = 7 + 4 = 11, dist G = 10 No Change

	B	C	D	E	F	G
status	√	√	√	x	√	x
link	A	B	C	D	C	D
dist	3	5	6	8	7	10

example of the loop invariants
Dist E = 8 and E has not been picked yet. This means that of all paths between A and E going through B or C or D or F (the picked vertices), the shortest is 8 (that shortest path is EDCBA). But there may be a shorter path going through the as yet unpicked vertex G.

Dist F = 7 and F *has* been picked. This means that of all paths between A and F going through B or C or D (the picked vertices), the shortest is 7; the path itself is FCBA. And furthermore this is *the* overall shortest path; there is no shorter path even if you go through the unpicked vertices E or G.

round 5 Pick E and update
dist E + EG = 8 + 1 = 9, dist G = 10 Change

	B	C	D	E	F	G
status	√	√	√	√	√	x
link	A	B	C	D	C	E
dist	3	5	6	8	7	9

round 6 Pick G and you're finished

At the end of the algorithm here are the conclusions.

(4) For each vertex V, dist V is the shortest distance from A to V.

The last table gives final results for all vertices.
For example, dist G = 9 so the shortest distance from A to G is 9.

(5) And you can use the links to find the shortest path itself:
link G = E, link E = D, link D = C, link C = B, link B = A.

A shortest path between A and G is A B C D E G.

warning
No adding is necessary at the end of Dijkstra's algorithm (it's *different* from Prim). In example 2, the shortest path between A and G is ABCDEG and the shortest distance between A and G is 9, *not*
9 + 8 + 6 + 5 + 3.

mathematical catechism (you should know the answer to these questions)

question 1 After any round of Dijkstra's algorithm for shortest paths from A, what does dist Q represent if Q is an unpicked vertex.

answer Dist Q is the length of the shortest path between A and Q using picked vertices as the only possible intermediates. (Dist Q may improve as more vertices get picked.)

> **warning** The answer is not shortest path "so far" or "up to now" or "at this time" because that's too vague.

question 2 After any round of Dijkstra's algorithm for shortest paths from A, what does dist Q represent if Q is a picked vertex.

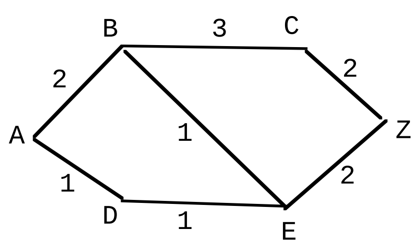
answer It is still the length of the shortest path between A and Q using picked vertices as the only possible intermediates. But more important, it is the length of *the* shortest path between A and Q (shortest of all possible paths); there is no shorter path no matter what intermediates you use.

PROBLEMS FOR SECTION 3.3

1. Make up an example to show that there can be *two* (equally) shortest paths from A to Z even if the edge weights are all different.

2. A graph has vertices A, B, C, D, E, P, Q, X, Z.
After round 4 of Dijkstra's algorithm for finding shortest paths from Q to the other vertices, vertices D, F, X, Z are picked. The other vertices are unpicked.
(a) If dist P = 8, what does that signify (quote the loop invariant).
(b) If dist X = 7, what does that signify

3. (a) Use Dijkstra's algorithm (but without formally keeping track of status, link, dist) to find the shortest paths from B to the other vertices in the diagram below. List all the shortest paths and their lengths and draw the tree of shortest paths
(b) For comparison, use Prim's algorithm starting with B to find a minimal spanning tree.



4. For the graph in problem 3, find shortest paths from A to the other vertices, keeping track of status, link, dist at each step.

5. For the given weight matrix, keep track of status, dist, link to find
(a) a shortest path from C to E
(b) shortest paths from B to all the other vertices

	A	B	C	D	E	F	G
A							
B	3						
C	6	2					
D	∞	4	1				
E	∞	∞	4	2			
F	∞	∞	2	∞	2		
G	∞	∞	∞	4	1	4	

6. For the given weight matrix, keep track of status, dist, link to find shortest paths from A to D.

	A	B	C	D	E	F	G	Z
A								
B	2							
C	∞	2						
D	∞	2	∞					
E	∞	4	3	4				
F	1	∞	∞	3	∞			
G	∞	∞	∞	∞	7	5		
Z	∞	∞	1	∞	∞	∞	6	

7. Look at Dijkstra's algorithm for shortest paths from A.

(a) True or False (and explain).
If vertex Z is picked right after vertex B then in the final table dist B ≤ dist Z.

(b) True or False (and explain). If Z is picked after B but not right after; say B then Q then Z are picked then in the final table dist B ≤ dist Z.

8. Here is the result of four rounds of Dijkstra's algorithm for finding shortest paths from vertex B to other vertices in a weighted graph.

vertex	A	C	D	E	F	G
status	√	√	√	x	√	x
link	B	B	C	D	C	D
cost	3	2	3	5	4	7

(a) What conclusions can you draw about the shortest distance between G and B and about the shortest path itself.
(b) What conclusions can you draw about the shortest distance between F and B and about the shortest path itself.

SECTION 3.4 THE RUNNING TIME (TIME COMPLEXITY) OF AN ALGORITHM

example 1 (running time of an algorithm which finds the largest [smallest] of n numbers)

Start with the n numbers $x_1, \ldots, x_n$. Here's an algorithm to find the maximum.

Compare x_1 with x_2 and choose the largest.
Compare the winner of the preceding round with x_3 and choose the largest.
Compare the winner of the preceding round with x_4 and choose the largest.
$\vdots$
The final step is a comparison of the latest winner with x_n.

With this algorithm it takes n−1 comparisons to find the largest (smallest) of n numbers it can be shown that no algorithm can do better.

The algorithm must perform other operations besides comparisons (e.g., it has to store the winner at the end of each round) but it performs comparisons more than any other operation so it's an appropriate unit of measure. We say that the algorithm has *running time* or *time complexity* n−1. The functions n−1 and n have the same order of magnitude (coming next) so for simplicity we say that the running time is n.

order of magnitude

Suppose f(n) and g(n) both approach ∞ as n → ∞ so that

$$(1) \qquad \lim_{n\to\infty} \frac{f(n)}{g(n)} = \frac{\infty}{\infty}$$

If the limit in (1) turns out to be ∞ then f(n) is said to be of a *higher order of magnitude* than g(n); i.e., f grows faster than g.
If the limit is 0 then f has a *lower order of magnitude* than g.
If the limit is a positive number then f and g have the same *same order of magnitude.*

The pecking order in (2) shows some well-known functions which approach ∞ as n → ∞, and lists them in increasing order of magnitude, from lower to higher.

$$(2) \qquad \boxed{\ln n,\ \sqrt{n},\ n,\ n^2,\ n^3,\ \ldots,\ (1.5)^n,\ 2^n,\ 3^n, \ldots,\ n!}$$

The list is not intended to be complete. There are functions with lower order of mag then $\ln n$, higher than n!, in between $\sqrt{n}$ and n, etc.

Here's the check that n^4 has a higher order of mag then n^3:

$$\lim_{n\to\infty} \frac{n^4}{n^3} = \frac{\infty}{\infty} = \lim_{n\to\infty} n \text{ (cancel)} = \infty$$

Here's the check that e^x has a higher order of mag then x^3:

$$\lim_{x\to\infty} \frac{e^x}{x^3} = \frac{\infty}{\infty} = \lim_{x\to\infty} \frac{e^x}{3x^2} \text{ (L'Hopital's rule)} = \frac{\infty}{\infty}$$

$$= \lim_{x\to\infty} \frac{e^x}{6x} \text{ (L'Hopital)} = \frac{\infty}{\infty}$$

$$= \lim_{x\to\infty} \frac{e^x}{6} \text{ (L'Hopital)} = \frac{\infty}{6} = \infty$$

On the other hand look at n^3 versus $5n^3$:

$$\lim_{n\to\infty} \frac{5n^3}{n^3} = \frac{\infty}{\infty} = \lim_{n\to\infty} 5 \text{ (cancel)} = 5$$

So $5n^3$ and n^3 have the same order of magnitude.

Intuitively, n^4 has a higher order of mag than n^3 because for *large* n, n^4 is *much* larger than n^3. In particular it is larger by a factor of n, a very big factor when n is large. On the other hand, $5n^3$ and n^3 have the same order of mag because for large n, $5n^3$ is not *much* larger than n^3. Of course it is larger by a factor of 5 but that factor stays constant as $n \to \infty$ so $5n^3$ does not decisively outdistance n^3.

Furthermore, not only do $5n^3$ and n^3 have the same order of mag but so do all of $5n^3$, n^3, $2n^3$, $n^3 + 2n - 3$, $8n^3 + n^2$ etc.

> A polynomial has the same order of magnitude as its highest power.
> For example,
>
> $n^4 + 3n^2$,
> $6n^4$
> $2n^4 - 7n^3$
> etc
>
> all have the same order of magnitude as n^4.

tractable versus intractable problems

Start with a problem of "size" n (e.g., find the smallest of n numbers, find an Euler cycle in a graph with n vertices, invert an $n \times n$ matrix). Suppose the problem can be solved in polynomial time, i.e., there is an algorithm to solve the problem which has a running time of n or n^2 or n^3 or n^4 etc. Then the problem is called *tractable*. On the other hand if all algorithms to solve the problem have running times higher than polynomial (e.g., exponential running time) then the problem is called *intractable*; for large n it would take centuries to solve the problem

big oh notation

If the running time of an algorithm is *less than or equal to* say n^4 then we say that the running time is $O(n^4)$ (big oh of n^4)

Often you can't pin down the precise running time of an algorithm because it might run faster for an easy problem of size n (e.g., a graph with n vertices and very few edges) as opposed to a messy problem of size n (a graph with n vertices and many edges). In that case you can find the running time in the worst situation and claim that the running time is O(worst)

example 2 (running time for Warshall's algorithm)

Let the adjacency matrix M be $n \times n$.
Use comparisons (is an entry a 1?) and additions as the units of measure.

Going from M_0 to M_1 takes one comparison for each of the n rows (to see if the entry in col 1 is a 1) for a total of n comparisons. And *at worst*, row 1 is added to each row which takes n additions for each of n rows, for a total of n^2 additions. So the total number of operations is $n^2 + n$.

To work your way up to M_n from M_0 you have to do these $n^2 + n$ operations n times for a final total of $n(n^2 + n) = n^3 + n^2$ ops. So, *at worst*, the running time is n^3, i.e., it's $O(n^3)$

warning

Don't say the running time of Warshall's algorithm is $O(n^3 + n^2)$. Use the simplest function with the same order of magnitude as $n^3 + n^2$ and call the running time $O(n^3)$. And use the big oh notation because n^3 was a worst case scenario.

example 3 (running time for Prim's algorithm)

Let M be an $n \times n$ weight matrix. Use the comparison as the unit of measure.

First count the comparisons you need in each round to find the unpicked vertex with the smallest cost. In round 1 there are $n-1$ unpicked vertices, and with the algorithm in example 1, it takes $n-2$ comparisons. In round 2 there are $n-2$ unpicked vertices and it takes $n-3$ comparisons to pick the smallest etc. In the $(n-2)$th round there are 2 unpicked vertices and it takes 1 comparison.

The total number of comparisons is

$$(n-2) + (n-3) + \ldots + 1$$

for reference (to be given out on exams if needed)

(A) $1 + 2 + 3 + \ldots + n = \dfrac{n(n+1)}{2}$

(B) $1^2 + 2^2 + 3^2 + \ldots + n^2 = \dfrac{n(n+1)(2n+1)}{6}$

To find the sum, use (A) but with $n-2$ playing the role of n:

$$(n-2) + (n-3) + \ldots + 1 = \frac{(n-2)(n-1)}{2}$$

Now count the comparisons necessary to update in each round (is $vv^* <$ cost v?) In round 1, after picking a vertex there are $n-2$ vertices to update so there are $n-2$ comparisons. In the next round there are $n-3$ comparisons etc. so the number of comparisons for updating is $(n-2) + (n-3) + \ldots + 1$ again.

The final total is $2 \cdot \dfrac{(n-2)(n-1)}{2} = (n-2)(n-1)$ so the order of mag of the running time is n^2. I don't call it $O(n^2)$ because the conclusion was that the order of mag *is* n^2, not just $\le n^2$.

$O(n^2)$ and $O(n^3)$ versus plain n^2 and plain n^3

When I counted operations for Prim's algorithm I found that for *any* $n \times n$ matrix the running time is n^2. For Warshall's algorithm I found that in a *worst* case the running time is n^3; for *some* matrices it could be $< n^3$.

So we say that Warshall's algorithm is $O(n^3)$ and Prim's algorithm is n^2 (without the big oh).

example 3 (running time for Dijkstra's algorithm)

Let M be an $n \times n$ weight matrix. Use the comparison as the unit of measure.

Dijkstra's algorithm must have the same running time as Prim's algorithm, namely n^2. The only difference between the algorithms is in *what* is being compared in the updating step (vv^* vs. cost v in Prim, dist $v^* + v^*v$ vs. dist v in Dijkstra). There is no difference in the *number* of comparisons in updating or picking.

PROBLEMS FOR SECTION 3.4

1. Here's the Bubblesort algorithm which arranges n numbers $x_1, \ldots, x_n$ in increasing order.

Find the larger of x_1 and x_2 and store the larger in x_2, the smaller in x_1.
Find the larger of x_2 and x_3 and store the larger in x_3, the smaller in x_2.
Continue until you find the larger of x_{n-1} and x_n and store the larger in x_n, the smaller in x_{n-1}.
By the end of this round, the largest of $x_1, \ldots, x_n$ has bubbled up to register x_n.
Now repeat the process with $x_1, \ldots, x_{n-1}$.
And repeat with $x_1, \ldots, x_{n-2}$ etc.

Find the running time of the algorithm using the comparison as the unit of measure.

2. (a) Find the running time of the following algorithm using the comparison as the unit of measure

```
Let k = 1,..., n
      Let i = 1,..., n
               Let j = 1,..., n
               If W(i,k) + W(k,j) < W(i,j), set W(i,j) = W(i,k) + W(k,j)
```

footnote
For the purposes of this problem it doesn't matter what the algorithm is for but in case you're curious, this is Warshall's algorithm for shortest distances (not the Warshall from Section 3.1 which finds a reachability matrix). If W is the weight matrix of a graph then at the end of the algorithm, W(i,j) is the shortest distance from v_i to v_j.

(b) Does it change the answer to part (a) if you use comparisons *and* additions as the unit of measure?

(c) Here's a totally meaningless algorithm. Find its running time using the comparison as the unit of measure.

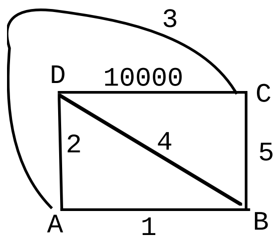
```
Let k = 1,..., n
    If W(1,k) + W(2,k) ≤ W(3,k) set W(1,1) = 4

Let i = 1,..., n
    If W(i,1) + W(i,2) ≤ W(i,3) set W(1,1,) = 5

Let j = 1,..., n
    If W(j,j) ≤ W(1,1) set W(1,1) = 6
```

3. A traveling saleswoman who has to visit cities $v_1, \ldots, v_n$ in her district wants a cycle, called a *Hamilton* cycle, which passes through every vertex exactly once (as opposed to an Euler cycle which passes through every edge exactly once). Furthermore she wants a Hamilton cycle with minimum distance so that she does a minimum of driving.

For example, for the weighted graph in the diagram, one of several Hamilton cycles is ABCDA with a total weight of 10008. The minimal Hamilton cycle for the graph in the diagram happens to be ADBCA, with weight 14.



Here's a (greedy) algorithm called the nearest neighbor algorithm which is reputed to usually produce a reasonable although not necessarily minimal Hamilton cycle for a weighted graph.

Start with any vertex, say x.
Pick the vertex, call it y, linked to x by the smallest edge and start a path with edge xy.
Look at all the vertices not picked yet and find the one, call it z, linked to y by the smallest edge. Grow the path by adding edge yz.
The path keeps growing in this fashion until it passes through all the vertices.
Then make a cycle by adding the edge from the last vertex picked back to x.

In other words, the path grows (greedily) one edge at a time from x to its nearest neighbor y, then to y's nearest neighbor z, then to z's nearest neighbor etc. Once the last vertex, say q is picked, close up the cycle with edge qx.

For the graph in the diagram, starting at vertex A, the nearest neighbor algorithm produces the Hamilton cycle ABDCA; its weight is 10008, not very close to the minimum so being greedy didn't get a good result this time.

Find the running time of the nearest neighbor algorithm for a graph with n vertices, assuming there are no multiple edges. (Two vertices which aren't connected can be thought of as joined by an edge with weight ∞, so you can consider that the graph is the complete graph K_n, with weights.)

4. Dijkstra's algorithm finds shortest paths from a fixed vertex to every other vertex. Suppose you want to use Dijkstra's algorithm over and over to find paths from *every* vertex to every other vertex. What's the running time of the over-and-over-again Dijkstra.

5. John claims that a certain algorithm is $O(n^2)$ and Mary says it's $O(n^3)$

(a) Could both be right.
(b) Could John be wrong and Mary right.
(c) Could John be right and Mary wrong.

6. In an $n \times n$ matrix, call row i and col i a *good* pair if all their entries are positive. For example, if

$$M = \begin{bmatrix} 1 & 2 & 3 & 5 \\ 4 & 2 & \pi & 6 \\ 1 & 6 & 8 & 1 \\ -2 & 9 & 2 & 7 \end{bmatrix}$$

then there are two good pairs, row 2 & col 2 and row 3 & col 3.
Here's an algorithm which finds all the good pairs.

Go across row 1 and ask "is the entry positive".
If all yeses, then go down col 1 and ask "is the entry positive".
If all yeses then you have found a good pair.
Repeat for each of the n rows of the matrix.

Choose a unit of measure and find the running time of my algorithm.

7. Consider an algorithm to solve a task of size n (e.g., a graph problem with n vertices).

(a) If the algorithm has running time n^2 [operations] and each operation takes 10^{-6} seconds what's the largest size task that can be done in one minute.

(b) If the algorithm has running time n^3 [operations] and each operation takes 10^{-6} seconds what's the largest size task that can be one in one minute.

8.(a) Suppose you want to solve a problem of size n. You have a computer and an algorithm with running time n^3. If you had a choice of getting a computer which is 10 times as fast or an algorithm with running time n^2 which would you prefer if
(a) n = 1000
(b) n = 10
(c) n = 9

SECTION 3.5 NETWORK FLOWS

transport networks

Fig 1 shows a transport network with a flow. The network is a weighted directed graph with no loops. The underlying undirected graph is connected. There is one source (vertex with exits but no entrances) named A, and one sink (entrances but no exits) named Z. The first weight on an edge, always positive, is the carrying capacity of the edge. The second weight is non-negative and is the actual flow on the edge (if there is no second weight it's assumed to be 0). In Fig 1, edge CZ has capacity 4 and flow 2 (say in gallons).

The flow through an edge can't exceed the edge capacity and there must be conservation of flow at a vertex in the following sense.

(1) For any vertex other than A and Z, the flow in equals the flow out

For example, at B in Fig 1 the flow in (from A and E) is 1 + 1 = 2 and the flow out (to C) is 2.

value of the network flow

(2) It can be shown that in the flow out of A equals the flow into Z.

This quantity is called the *value of the network flow*.
For example, in Fig 1 the flow out of A (to B and D) is 1 + 3 = 4 and the flow into Z (from C and E) is 2 + 2 = 4; the value of the network flow is 4.

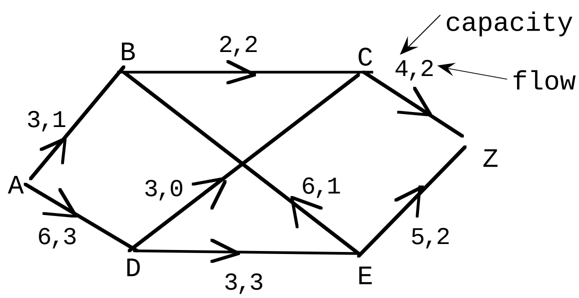


FIG 1

cuts

If the vertices of the network are divided into two groups, one containing A and the other containing Z, then the set of all edges from one group to the other is called a cut (Fig 2).

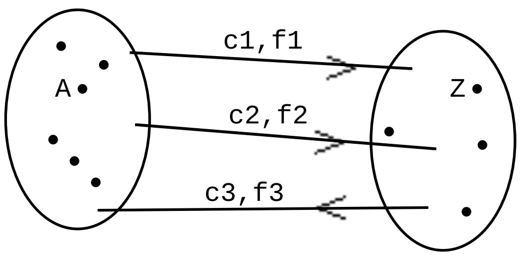
The *capacity* of the cut in Fig 2 is $c_1 + c_2$, *the amount that can be carried from the A side to the Z side*.

The *cut flow* in Fig 2 is $f_1 + f_2 - f_3$; *add the flow going from the A side to the Z side and subtract anything going the other way*.

warning

To get the cut *capacity*, IGNORE the edges going the "wrong" way (i.e., from the Z-side to the A-side)
To get the cut *flow,* SUBTRACT the edges going the wrong way.

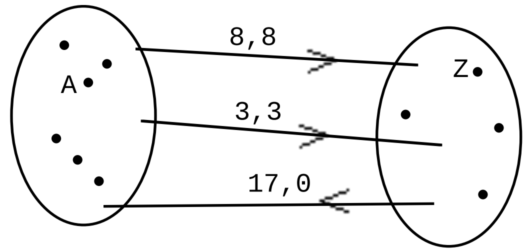
A cut is *saturated* if the cut flow equals the cut capacity (Fig 3), i.e., if the edges from the A side to the Z side carry max flow and the edges from the Z side to the A side carry zero flow.



cut capacity $c_1 + c_2$

cut flow $f_1 + f_2 - f_3$

FIG 2



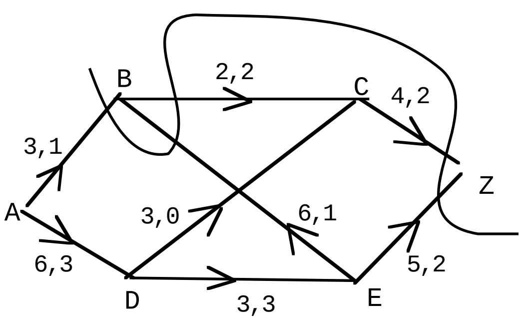
saturated cut

FIG 3

(3) *It can be shown that all cuts carry the same flow which in turn is the same as the value of the network flow itself.* (This is a generalization of (2).)

Intuitively, the network flow must be carried across every cut since however you choose to divide up the vertices to form a cut, the network flow must get from the A-side to the Z-side.

Fig 4A shows Fig 1 again along with the cut separating A, D, E, C from B, Z.



cap = AB + EB + CZ + EZ
= 3 + 6 + 4 + 5
= 18

flow = AB + EB + CZ + EZ - BC
= 1 + 1 + 2 + 2 - 2
= 4

FIG 4A

Fig 4B shows some more cuts in the network of Fig 1.
All the cuts have flow 4. And the network flow is 4.

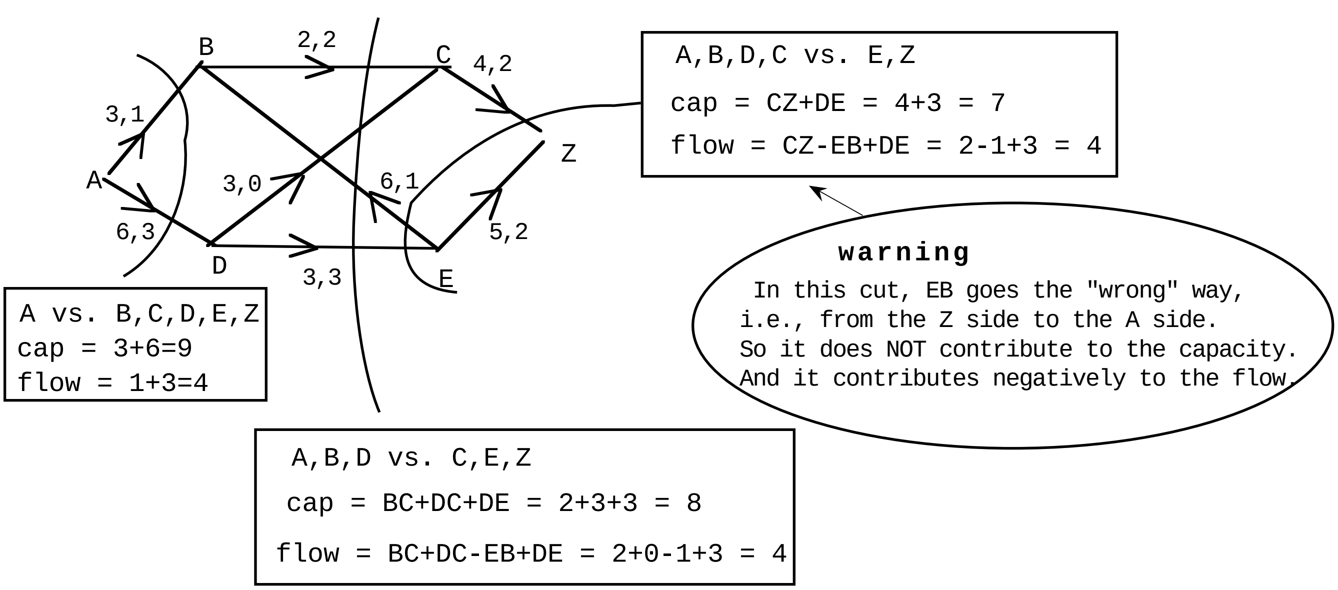


FIG 4B

max flow

Look at a transport network and all the possible cuts. By (3), the network flow must go across every cut, so *once some cut is saturated, the max network flow is achieved*.

(Furthermore *the maximum flow possible through the network equals the minimum of all the cut capacities*---can't do better than the weakest link.)

Fig 5 shows one possible max flow---you know it's max because of the saturated cut. The purpose of this section is to show you how to get it.

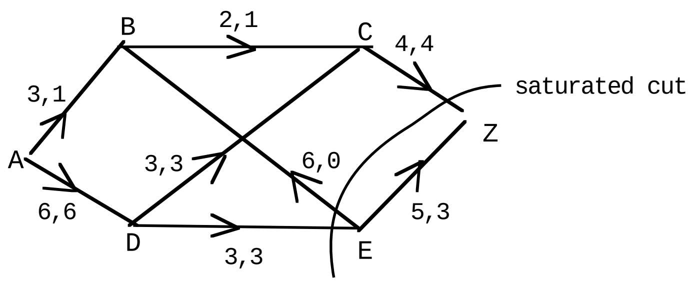


FIG 5 max flow

preview of the labeling algorithm for finding a max flow

Suppose you have an initial flow (maybe all zero). To get a max flow, aim to increase the flow until some cut is saturated. In a small diagram you can do it by inspection. The labeling procedure is designed to do it mechanically.

Look at the network in Fig 6.

By inspection you can increase the flow by sending 4 gallons on path ADZ (can't send more on this particular path since edge DZ only has room for 4 more gallons).

Here's how to get the corresponding result by a labeling procedure instead of inspection.

Give A the label $(\cdot,\infty)$ to indicate that A is a bottomless source (Fig 7).

Once A is labeled, you can label D by thinking as follows: A can supply any amount, edge AD can carry 6 more to D, so give D the label $(A^+,6)$ to indicate that D can get 6 gallons from A (Fig 7).

Once D is labeled you can label Z: The label on D indicates that D can supply 6 more, edge DZ can carry 4 more to Z, so give Z the label $(D^+,4)$.

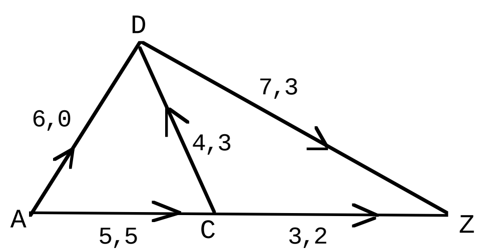


FIG 6 Initial flow

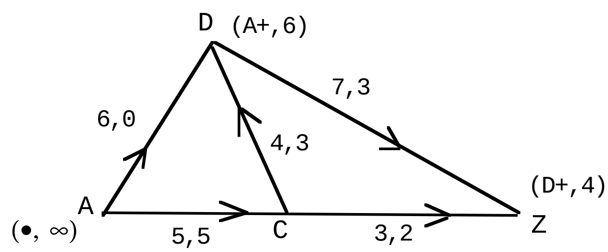


FIG 7 Round 1 Labels

Once Z is labeled, trace the labels from Z back to A and augment the flow on the path ZDA by 4, the amount on the Z-label (Fig 7A).

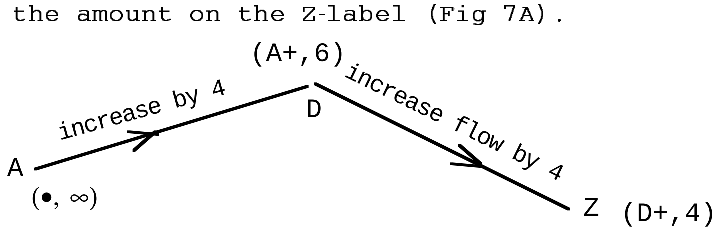


FIG 7A

This produces the new flow in Fig 8, a result you saw earlier by inspection. But now you have a method that works on a large, not easily-inspected network.

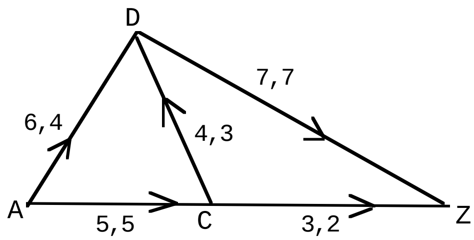


FIG 8 Round 1 Augmented flow

Let's try to increase the flow again
By inspection of Fig 8 you can't send more to Z conventionally on paths ACZ, ADZ and ACDZ since edges AC and DZ are saturated. But we can send 1 gallon *less* from C to D and keep the flow at D conserved by taking an extra gallon from A to replace it. Then C will have an extra gallon to send to Z, increasing the network flow.

Here is the labeling process corresponding to this idea.
Begin with the A-label $(\cdot,\infty)$ again (Fig 9).
Give D the label $(A^+,2)$ since D can get 2 extra gallons from A.
Once D is labeled, you can give C a label as follows: Right now, C is sending 3 gallons to D but 2 of the gallons can be *retracted* since the D label shows that D can get 2 more from A to make up the loss. So give C the *negative* label $(D^-,2)$.
Once C is labeled you can label Z as follows: The C label indicates that C can get 2 extra gallons, edge CZ can carry 1 more to Z, so give Z the label $(C^+,1)$.
Now that Z is labeled, trace the labels back to A: Change the flow on each edge of the path ZCDA by 1, the amount on the Z label. The negative label $(D^-,2)$ on vertex C means that the flow on edge CD should be *decreased* by 1. The other edge flows are *increased* by 1. Fig 10 shows the augmented flow. (Note that we still have conservation of flow at every vertex.)

Since this network is small, you can probably spot a saturated cut in Fig 10 by inspection and realize that the flow in Fig 10 is maximum. But you can get a saturated cut automatically by continuing the algorithm. In Fig 11, I found labels for A, D, C but got stuck trying to label Z. When you get stuck in the labeling process the cut separating the labeled vertices from the unlabeled (just Z here) will be saturated, indicating that the flow is maximum and the algorithm is over. The max flow is in Fig 11. The value of the max flow is 10 (the flow across the saturated cut -- also the flow out of A and also the flow into Z).

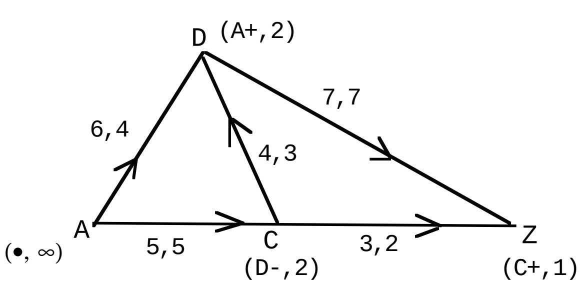


FIG 9 Round 2 Labels

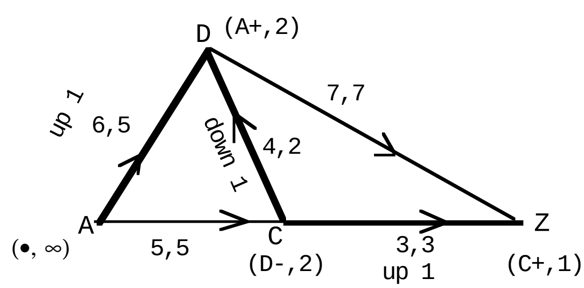


FIG 10 Round 2 Augmented flow

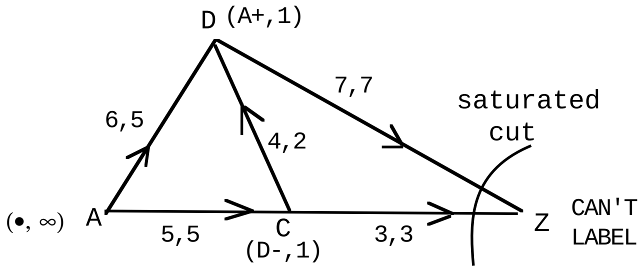


FIG 11 Round 3 max flow

the labeling algorithm in general

Here are the general steps in the algorithm.

Begin with the A-label $(\cdot,\infty)$

Then continue labeling using the rules illustrated in Figs 12 and 13. The aim is to label a path from A to Z. The labeling process is not unique; some networks can be labeled in more than one way.

Fig 12 illustrates the rule for assigning positive labels.

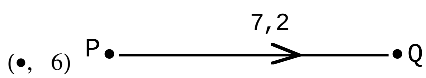


FIG 12 Q gets label (P+, 5)

In Fig 12, P's label indicates that P can get 6 more gallons (doesn't matter from where).

Edge PQ has room to carry 5 more to Q.

Take the smaller of 5 and 6 and give Q the label $(P^+, 5)$ to indicate that Q can get 5 more from P.

In general, a vertex Q can get a positive label when an edge leads from a labeled vertex to Q and that edge is not saturated.

Fig 13 illustrates the rule for assigning negative labels.

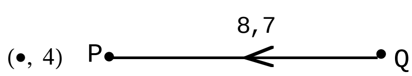


FIG 13 Q gets label $(P^-, 4)$

In Fig 13, Q can take back 4 of the 7 it is sending to P because P's label indicates it can get 4 extra gallons to make up for a loss.

So give Q the label $(P^-, 4)$

(The amount on the Q label is the smaller of the flow 7 on edge QP and 4 on the P label but it's better if you understand it rather than memorize something.)

In general, a vertex Q can get a negative label when an edge leads from Q to a labeled vertex and that edge carries nonzero flow (which can be retracted)

Here are the two typical situations where Q can't get a label.

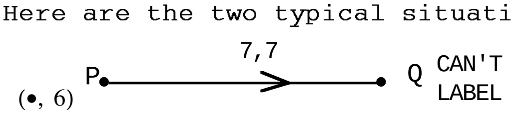


FIG 14

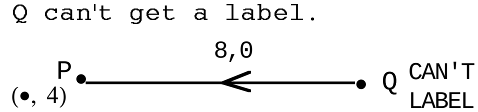


FIG 15

In Fig 14, Q can't get a positive label because edge PQ is filled.

In Fig 15, Q can't get a negative label because Q is sending zero gallons to P (there is nothing to retract).

Here's how to use the labels to augment the flow.
Once Z is labeled, trace the labels back to A and augment the flow (either up or down) by the amount on the Z-label as illustrated in Fig 16.

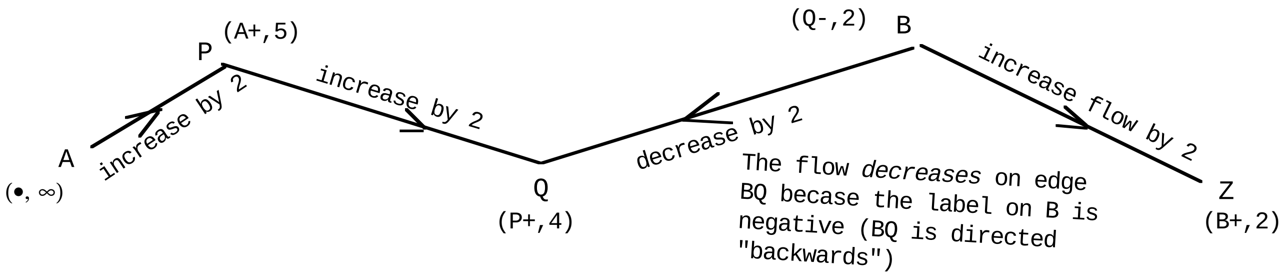


FIG 16 Augmenting the flow

warning

Vertex Q in Fig 16 has label $(P^+,4)$ but the flow is augmented on edge PQ by 2, not 4. The amount of the change is determined by the Z-label, $(B^+,2)$, and all edges are augmented by the same amount (some up, some down).

The algorithm continues until the labeling process gets stuck before reaching Z. In that case the cut separating the labeled vertices from the can't-be-labeled vertices is saturated and the flow is max.

why the algorithm works

Here's why the cut between the labeled and the can't-be-labeled vertices must be saturated.
In Fig 17, if say pipeline XR weren't filled to capacity then R could get a positive label (contradiction) and if say edge TY weren't carrying zero flow then T could get a negative label (contradiction). So XR carries as much as it can and TY carries zero flow.

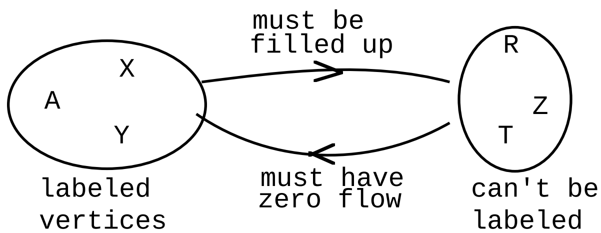


FIG 17

example 1

I'll continue from Fig 18 and find a max flow, a saturated cut to verify that it's a max flow, and the actual value of the max flow.

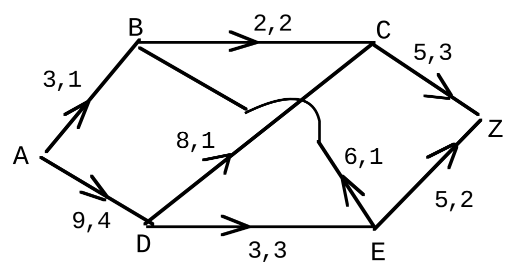


FIG 18

round 1

After A gets the label (·, ∞) here's one way to continue.

There is room on edge AD for 5 more gallons and A has infinitely many gallons available so D can get label (A+, 5) (Fig 19a)

There is room on edge DC for 7 more gallons but D can supply only 5 so C gets label (D+, 5).

There is room on edge CZ for 2 gallons and C can supply 5 so Z gets label (C+, 2).

> *footnote*
> Can also give B the label (A+,2), E the label (B−,1) and Z the label (E+,1). But I'll just use the labels I have in Fig 19a.

Follow the labels back from Z to A (Fig 19b). On the path ZCDA, augment each flow by 2, the amount on the Z label (Fig 19b). All flows *increase* because all the labels are positive

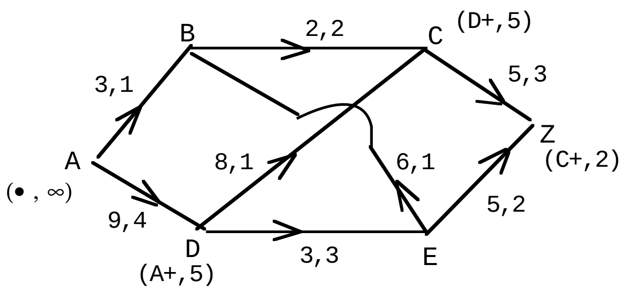


FIG 19a Label a path to Z

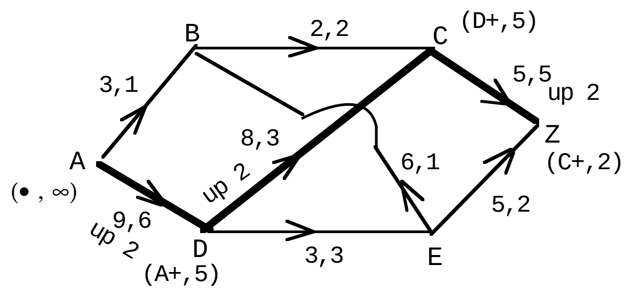


FIG 19b Augment along that path

round 2

Continue from Fig 19b.

A gets label (·, ∞).

There is room on edge AB for 2 gallons so B gets label (A+, 2)

Similarly D gets the label (A+, 3)

Even though B has a label, C can't get a B label since edge BC is full but C can get a D label. Doesn't help since edge CZ is saturated.

Since B can get 2 gallons from A, E can take back the gallon it is sending to B so E can get the negative label (B−, 1)

There is room on edge EZ for 3 gallons but E can supply only 1 so Z gets label (E+, 1) (Fig 20a).

On the path ZEBA (Fig 20b) augment the flow by 1, the amount on the Z label (Fig 20b). The flow on edge EB goes *down* by 1 because E has a negative label (this *increases* the flow from A to Z since edge EB goes the "wrong" way.)

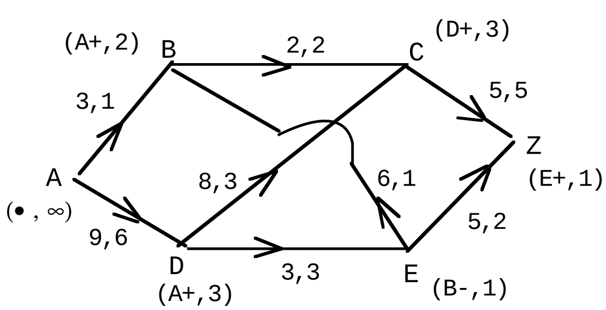


FIG 20a Label a path to Z

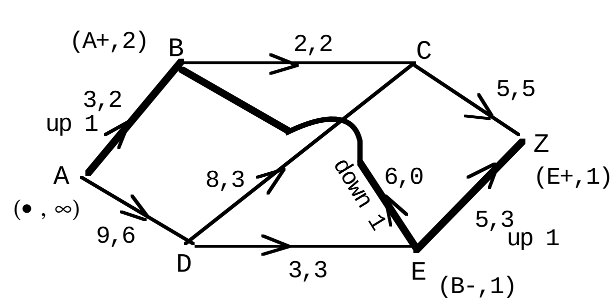


FIG 20b Augment along that path

round 3

I labeled everything I could but couldn't give E and Z labels.
The cut separating E and Z from the other vertices is saturated and the algorithm is over.

Fig 21 shows a max flow. Value of the max flow is 8, the flow on the saturated cut (also the flow out of A, also the flow into Z).

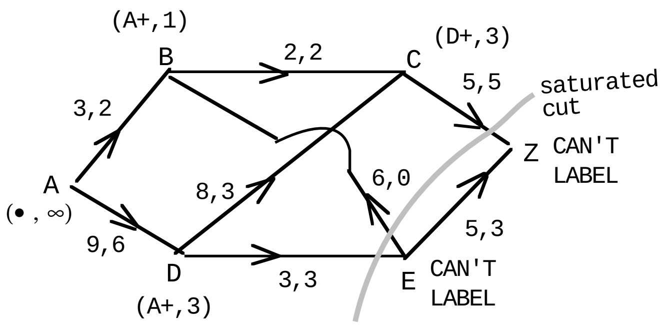
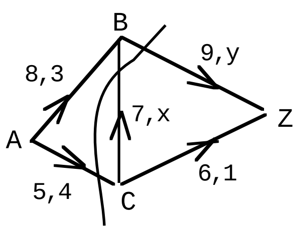


FIG 21 Try to label a path to Z

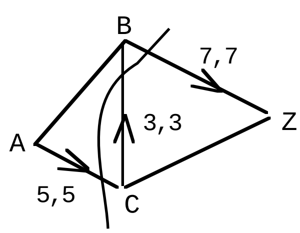
> **warning**
> The algorithm isn't finished *until you label as many vertices as you can but eventually get stuck* and can't label Z, in which case the saturated cut appears automatically as the *cut between the set of labeled vertices and the set of can't-label vertices*. If you find this or any other saturated cut by inspection then you are bypassing the algorithm. You can get away with this in small networks but not in large networks and not on exams.

PROBLEMS FOR SECTION 3.5

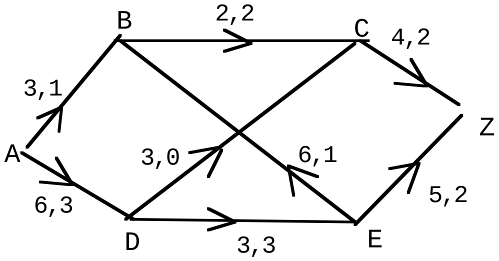
1. (a) Find flows x and y.
   (b) Find the capacity and flow across the indicated cut.
   (c) Find (effortlessly) the flow across every other cut.



2. Is the cut in the diagram saturated.

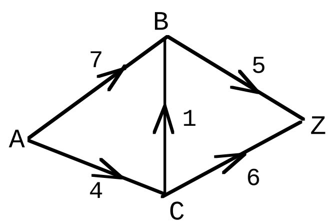


3. Look at the following network flow.

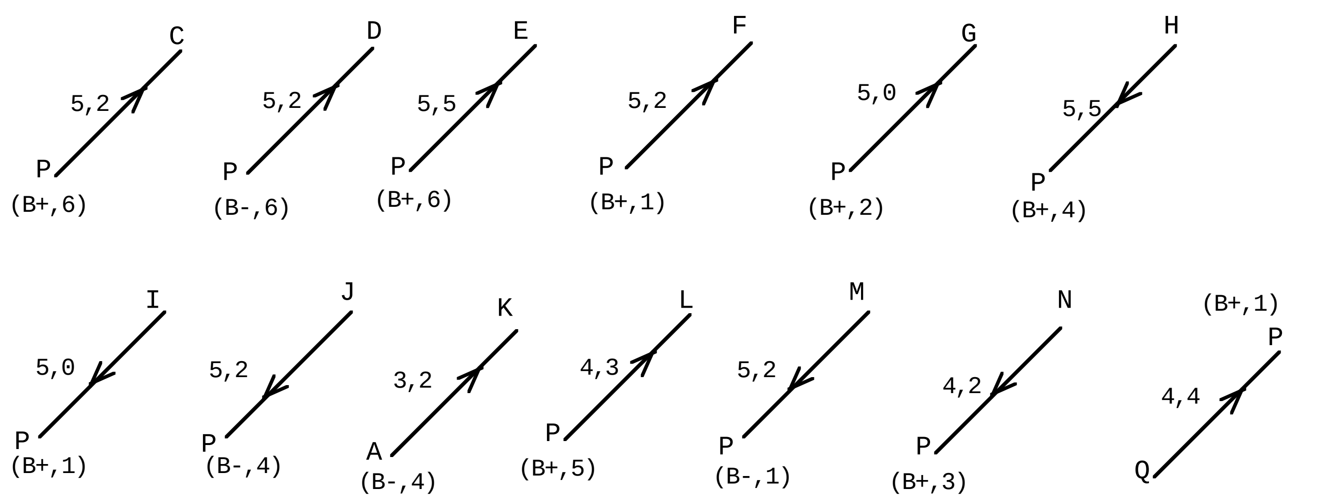


(a) Find the capacity and flow for the following two cuts.
  (i) A,B,D versus the other vertices
  (ii) A,E versus the others
(b) (a counting problem) How many cuts are there.

4. By inspection, find all cut capacities in the diagram below (the weight on each edge is the edge capacity), find a min cut and hence a max flow value for the network. And by inspection find an actual max flow.



5. Put down labels whenever possible.

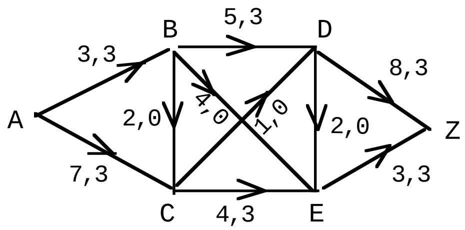


6. Can Z ever get a negative label.

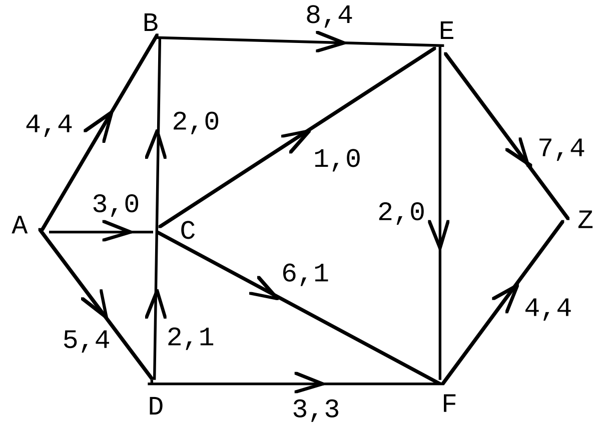
7. Go back to Fig 21 which says Z can't be labeled. There is room on edge EZ for 2 more gallons so why can't Z get the label (E^+, 2)?

8. Use the labeling algorithm to find a max flow and a saturated cut.
(Your work may differ from mine because labels are not unique but we should at least agree at the end on the *value* of the max flow.)

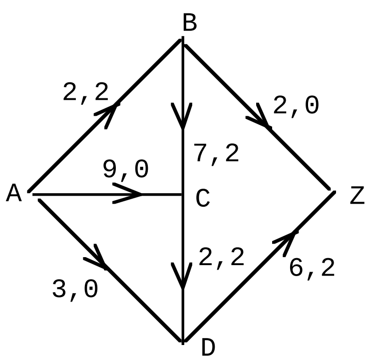
(a)



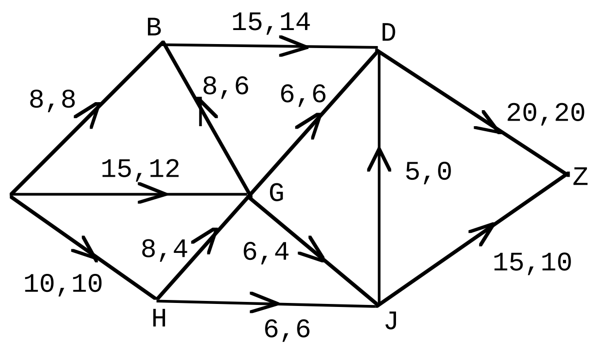
(b)



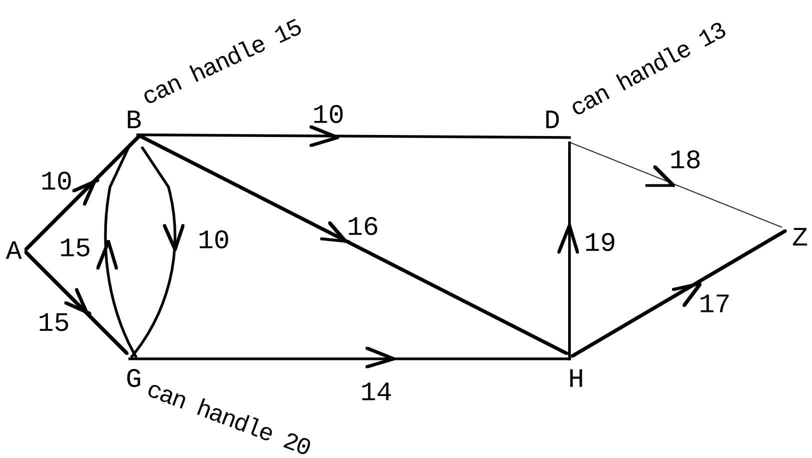
(c)



(d)



9. In the diagram below, in addition to *edges* with carrying capacities, several *vertices* have capacities as well; B can handle at most 15 gallons, G at most 20 and D at most 13 gallons.
So the diagram is not a transport network. But see if you can be clever and turn it into a transport network. (And then stop.The purpose of this problem is to illustrate that a problem that does not start out as a traditional network flow problem can be turned into one.)
Suggestion. Think of vertex B as an industrial complex with a receiving dock B_1 and a shipping dock B_2 where the road carrying items from B_1 to B_2 has capacity 15.



REVIEW PROBLEMS FOR CHAPTER 3

1. Given the distances in the chart below between cities B, E, F, G, I, S, T.
(a) Use the appropriate algorithm, to find a highway system that connects them minimally. How may miles in the minimal system.

(b) Suppose E and I must be directly connected. Find a minimal highway system with this restriction.

	B	E	F	G	I	S
E	119					
F	174	290				
G	198	277	132			
I	51	168	121	153		
S	198	303	79	58	140	
T	58	113	201	164	71	196

2. Here's the result of several rounds of Dijkstra's algorithm for shortest paths from A.

	B	C	D	E	F
status	√	√	x	x	√
link	A	B	F	C	A
dist	2	4	4	6	1

Fill in the blanks.
The shortest distance between A and E _______________ is _________.

3. Let M be the adjacency matrix of a digraph with 10 vertices v1,..., v10.
I've run Warshall's algorithm.

(a) (i) What can you conclude if there is a 1 in row 3, col 5 of M_4.
(ii) What can you conclude if there is a 0 in row 3, col 5 of M_4.

(b) Suppose there's a 1 in row 2, col 3 of M_6. Are these conclusions correct.
(i) There's a path from v_2 to v_3.
(ii) You can get from v_2 to v_3 without stepping on v_8.
(iii) You can get from v_2 to v_3 but you'll have to step on v_6 along the way.

(c) Suppose there's a 0 in row 2, col 3 of M_6. Are these conclusions correct.
(i) There's no path from v_2 to v_3.
(ii) There might be a path from v_2 to v_3 but if so you'll have to step on at least one of v_7, v_8, v_9, v_{10} along the way.

4. Start with a directed graph with n vertices. A vertex v in a digraph is called a *hub* if there is an edge from v to every other vertex and an edge from every other vertex to v.
(a) Make up an algorithm which examines the adjacency matrix of a digraph (without loops) with n vertices and finds all hubs. Describe your algorithm in simple English.
(b) Find the running time of your algorithm (after choosing an appropriate unit of measure).

5. Find M_∞ if

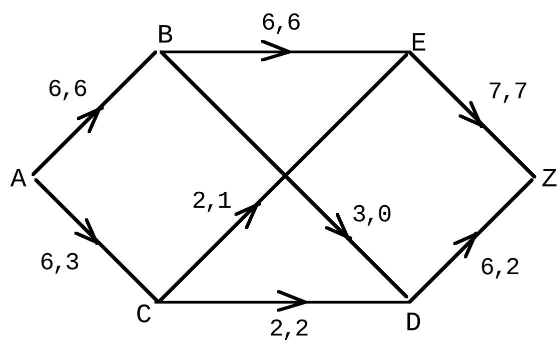
$$M = \begin{matrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{matrix}$$

6. Here's a weight matrix for a digraph and a table from round 1 of Dijkstra's algorithm for finding shortest distances from P. Go one round further and state whatever conclusions you can at that stage.

	A	B	C	D	E	F	G	P
A	0	∞	∞	∞	∞	∞	∞	∞
B	2	0	∞	∞	∞	∞	∞	∞
C	∞	1	0	∞	∞	∞	4	∞
D	∞	∞	3	0	∞	∞	8	∞
E	∞	∞	∞	∞	0	3	∞	∞
F	2	∞	∞	8	∞	0	∞	∞
G	∞	6	∞	∞	∞	∞	0	∞
P	10	∞	∞	∞	4	∞	∞	0

A	B	C	D	E	F	G
x	x	x	x	√	x	x
P	P	P	P	P	E	P
10	∞	∞	∞	4	7	∞

7. Use the appropriate algorithm to find a max flow and a saturated cut.



CHAPTER 4 RECURRENCE RELATIONS

SECTION 4.1 SETTING UP RECURRENCE RELATIONS TO DO COUNTING PROBLEMS

example 1

Look at binary strings (strings of 0's and 1's).

For example, 100111 is a binary string of length 6.

The problem is to count the number of binary strings of length n with no consecutive 0's.

For convenience let's call them "good" strings.

Let a_n be the number of good strings of length n (good n-strings).

Every good n-string (where $n \geq 3$) is exactly one of these two types:

(a) 1 followed by a good $(n-1)$-string

(b) 01 followed by a good $(n-2)$-string

For example, 11011101 is a good 8-string and fits into category (a) because it's a 1 followed by the good 7-string 1011101. And 0111011 is a good 8-string of type (b) because it's 01 followed by the good 6-string 110110.

There are a_{n-1} strings of type (a).

There are a_{n-2} strings of type (b).

Since the total number of good n-strings is number of (a)'s + number of (b)'s,

(1) $$a_n = a_{n-1} + a_{n-2} \quad \text{for } n \geq 3$$

And by inspection there are two good 1-strings (1 and 0) and three good 2-strings (11, 10, 01). So

(2) $$a_1 = 2 \quad \text{and} \quad a_2 = 3$$

You can put (1) and (2) together to grind out successive values of a_n:

$$a_3 = a_2 + a_1 = 3 + 2 = 5$$
$$a_4 = a_3 + a_2 = 5 + 3 = 8$$
$$a_5 = a_4 + a_3 = 8 + 5 = 13$$

In other words there are 5 good strings of length 3, 8 good strings of length 4, 13 good strings of length 5 and you can keep going as far as you like.

recurrence relations and initial conditions

The equation in (1) is called a *recurrence relation* or a *recursion relation* (rr) or a *difference equation* (ΔE).

The specific values in (2) are called *initial conditions* (IC).

The rr in (1) is *second order* because it tells you how to get a term from the preceding term and the pre-preceding term; i.e., it's second order because you need two IC to get rolling.

For example, the recurrence relation

$$y_n = 3y_{n-1} + n$$

is a *first order* rr because it tells you how to get a term from the preceding term; you only need one IC to get started.

Here are some examples of third order recurrence relations.

$$b_n = b_{n-3}$$
$$b_n = 6b_{n-1} + nb_{n-3}$$
$$b_n = 9b_{n-1} + 2\,b_{n-2} - 5b_{n-3}$$

In each case you'd have to know b_1, b_2 and b_3 (or b_0, b_1, b_2 or b_8, b_9, b_{10} etc.) before you could use the relation to keep finding successive b_n values.

example 1 continued

Suppose the strings can use any of the 10 digits, not just 0 and 1. How many strings of length n have no consecutive 0's.

solution Now every good n-string must be exactly one of these two types.

(a) $\overline{0}$ (nonzero) followed by a good (n−1)-string
(b) $0\overline{0}$ followed by a good (n−2)-string

There are $9a_{n-1}$ strings of type (a) because the $\overline{0}$ can be picked in any of 9 ways and the good (n−1)-string can be picked in a_{n-1} ways.

There are $9a_{n-2}$ strings of type (b). So

$$a_n = 9a_{n-1} + 9a_{n-2} \text{ for } n \geq 3$$

This is a second order rr so I need two IC:

$a_1 = 10$ (every string of length 1 is good)

$a_2 = 99$ (there are 100 two digit strings and they're all good except 00)

Then

$$a_3 = 9a_2 + 9a_1 = 891 + 90 = 981$$

$$a_4 = 9a_3 + 9a_2 = 8829 + 891 = 9720 \text{ and so on}$$

example 2

Let u_n be the number of words of length n that contain 3 or more A's in a row (good words). Find a recursion relation and IC for u_n.

solution A good word of length n must be exactly one of the following types:

(a) $\overline{A}$ followed by a good word of length n−1

warning 1

Don't call this $\overline{A}\ u_{n-1}$.

That confuses a d*escription* (good word of length n−1) with *how many* there are with that description (the number u_{n-1}).

(b) $A\overline{A}$ followed by a good word of length n−2
(c) $AA\overline{A}$ followed by a good word of length n−3
(d) AAA followed by *any* word of length n−3

warning 2

This type is AAA followed by *any* (n−3)-word, *not* AAA followed by a *good* (n−3)-word; and the number of words of type (d) is 26^{n-3}, not u_{n-3}.

For example, CZAAAAB is type (a), ABXYAAAZAAAA is type (b), AAXYAAABZ is type (c), AAAAA and AAACXYZ are type (d).

There are $25u_{n-1}$ words of type (a).

There are $25u_{n-2}$ words of type (b).

There are $25u_{n-3}$ words of type (c).

There are 26^{n-3} words of type (d).

So

$$u_n = 25u_{n-1} + 25u_{n-2} + 25u_{n-3} + 26^{n-3} \text{ for } n \geq 4$$

This is a 3rd order rr and needs three IC:

$u_1 = 0,\ u_2 = 0,\ u_3 = 1.$

Now you can start more computing values of u_n.

$u_4 = 25u_3 + 25u_2 + 25u_1 + 26^1 = 25 + 0 + 0 + 26 = 51$

$u_5 = 25u_4 + 25u_3 + 25u_2 + 26^2 = 1275 + 25 + 0 + 676 = 1976$

etc.

warning 3

When you claim that "a good n-word must be one of the following types" make sure that your types don't overlap (or else you'll overcount) and that they include all possibilities, i.e., are exhaustive (or else you'll undercount).

PROBLEMS FOR SECTION 4.1

1. You can go up a flight of stairs taking either one or two steps at a time. For example, you can climb a 6-step staircase with the patterns

1221, 2211, 1122, 111111, 222, etc

Let y_n be the number of ways of you can climb a staircase with n steps.

(a) Find a rr and IC for y_n.

(b) Use part (a) to find the number of ways to climb a 6-step staircase.

2. The problem is to count the number of words of length n with an even number of A's. For example, ZABBXA is a good word of length 6.
Find a recursion relation and IC.

3. A legal string consists of digits and the three operators +, *, − all arranged so that they make arithmetic sense.
The length of a string is the number of symbols in the string.
Here are some examples of legal strings:

2 3 4 + 8 1 * 0 + 1 2 3 4 5 − 29 (length is 17)
0 0 0 1 4 + 8 −3 (length is 9)

It's considered legal to have leading 0's so 000014 is OK.
Can't have two operations in a row so 2 + * 3 is illegal.
Every legal string must begin with a digit and end with a digit.

Let u_n be the number of legal strings of length n.
Every legal string with length 2 or more begins either with DD (two digits) or D op (a digit followed by an operation).
In particular, a legal n-string, $n \ge 3$, must be one of these two types:

(i) D followed by a legal (n−1)-string
(ii) D op followed by a legal (n−2)-string
(Stop and see if you could have figured this out for yourself.)

For example, the 9-string 346+098−2 is the digit 3 followed by the legal 8-string 46+098−2. The 9-string 3+456*1+0 is 3+ followed by the legal 7−string 456*1+0.

(a) Find a recursion relation and IC for u_n.

(b) Use the rr to find the number of legal strings of length 4.

4. A code uses the three words a, ab, bc. A message is a sequence of words. For example, one message is a ab ab bc a which we will write without spaces as aababbca.

Spaces aren't necessary between words in a message because this happens to be a uniquely decipherable code. For instance the message aababbca can be decoded in only one way, namely as a ab ab bc a.

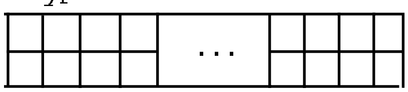
Some other messages for instance are a, aaa, ab, aab.

(a) Find a recursion relation and IC for the number of messages of length n (i.e., with n symbols).

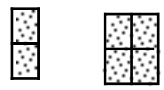
(b) Use part (a) to find the number of messages of length 9.

5. A board comes with two rows of squares.

The diagram shows the two types of tiles which can be used to cover it. (Tiles of the first type can be turned sideways.)

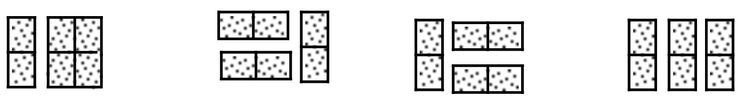


board



tiles

For example here are some ways to tile a board of length 3



Let y_n be the number of ways of tiling a board of length n (an n-board).

Find a recurrence relation and IC for y_n.

6. Sentences are composed from among the 26 letters A–Z and the symbol # representing a blank. A sentence can't begin or end with a blank or have two blanks in a row. For example, AA#BZZG#A is a 9-symbol sentence, HOW#ARE#YOU is an 11-symbol sentence.

Set up a recurrence relation with IC to find the number of n-symbol sentences.

7. Look at strings of 0's, 1's, 2's, 3's. A good string is one where a 3 never occurs past a 0; i.e., once a 0 appears the rest of the string can contain only 0's, 1's, 2's.

For example the string 123023 is *not* a good string because there is a 3 to the right of a 0.

Let a_n be the number of good n-strings (strings of length n). Find a recurrence relation and IC for a_n. Then use them to find the number of good strings of length 4.

8. Let b_n be the number of binary strings of length n with two or more consecutive 0's (e.g., 1000001, 0010).

(a) Look at this attempt.

The good strings of length n have to be one of these types

(i) 00 followed by any (n–2)-string

(ii) any digit followed by a good (n–1)-string

Derive the rr that goes with this attempt.

And then explain what's wrong with the attempt.

(b) Look at this attempt.

The good strings of length n have to be one of these types

(i) 00 followed by any (n–2)-string

(ii) 1 followed by a good (n–1)-string

Derive the rr that goes with this attempt.

And then explain what's wrong with the attempt.

(c) Get a rr, with IC, that works. Then find b_5

9. Call a string of A's, B's, C's, D's good if it has no consecutive identical letters.
For example ABCACACD and ABABABAB are good strings. The string BCCCAD is no good because of the CCC.

Let y_n be the number of good n-strings (i.e., number of good strings of length n).

(a) Find y_n directly using the methods of Chapter 1.

(b) Find a recurrence relation and IC for y_n.

(c) Check that your answer in (b) does satisfy the rr and IC from (a).

10. Let $S_n = \{1,2,3,4,...,n\}$.

Let y_n be the number of subsets of S_n that contain no consecutive integers (call them good subsets).

For example some good subsets of S_{10} are the null set

ϕ (the null set)
$\{1,3,5\}$
$\{2,4,7,10\}$
$\{5\}$.

Some *not*-good subsets are $\{3,4,7\}$, $\{1,5,6\}$.

Find a recurrence relation and IC for y_n. Then use them to find y_8.

Suggestion: Divide the good subsets of S_n into these two types: "contains n" and "doesn't contain n".

11. Your bank pays 6% interest a year, compounded monthly.
This means that the bank pays you 1/2% interest every month. The interest is added to your account so that the next month you get interest on your original principal plus the accumulated interest.

For example if you start with \$400 in the bank then at the end of a month you get $.005 \times 400 = \$2$ interest. At the end of the next month you'd get interest on \$402.

You deposit \$1000 initially and then starting with the next month, you deposit \$200 a month.

Let u_n be the amount of money you have after n months.

Find a recurrence relation and IC for u_n.

12. Let a_n be the number of ways in which 2n people (an even number of people) can be divided into n pairs. With this notation, a_8 is the number of ways of pairing up *sixteen* (not 8) people, a_7 is the number of ways of pairing up *fourteen* people

(a) Write down a formula for a_n immediately (using Chapter 1).

(b) What's wrong with the following way of getting a recurrence relation for a_n.

Pick a pair. Can be done in $\binom{2n}{2}$ ways. Now there are $2n-2$ people left. They can be paired up in a_{n-1} ways. So $a_n = \binom{2n}{2} a_{n-1}$.

(c) Get a recurrence relation and IC for a_n correctly.

(d) Check that the answer in (a) does satisfy the recurrence relation in (c).

13. You have n switches, in order from 1st to n-th, each of which can be ON or OFF. Here are the flipping rules.

The first switch can be flipped (ON to OFF or OFF to ON) any time.

But except for the first one, a switch can't be flipped (either way) until the preceding switch is ON and all the ones before that are OFF (e.g., you can't change the 10-th switch until the 9-th is ON and the first 8 are OFF).

For example, if you have 3 switches it takes 7 flips, as follows, to go from OFF OFF OFF to OFF OFF ON:

```
OFF OFF OFF
ON  OFF OFF
ON  ON  OFF
OFF ON  OFF
OFF ON  ON
ON  ON  ON
ON  OFF ON
OFF OFF ON
```

You start with n switches OFF and want to end up with the n-th switch ON and the others OFF. Let u_n be the number of flips it takes. Find a rr and IC for u_n.

SECTION 4.2 SOLVING HOMOGENEOUS RECURRENCE RELATIONS

equivalent forms of a recurrence relation

The three equations

$$y_n = 3y_{n-1} + 5y_{n-2}$$
$$y_{n+2} = 3y_{n+1} + 5y_n$$
$$y_{n+10} = 3y_{n+9} + 5y_{n+8}$$

are the *same* (second order) recurrence relation. Each says

term = 3·preceding term + 5·pre-preceding term

Similarly, these two recursion relations with IC are the same:

$$y_n = ny_{n-1} + 2^{n+3} \text{ for } n \ge 3 \text{ with IC } y_1 = 7,\ y_2 = 4$$
$$y_{n+1} = (n+1)y_n + 2^{n+4} \text{ for } n \ge 2 \text{ with IC } y_1 = 7,\ y_2 = 4$$

warning
In the second version, 2^{n+3} becomes 2^{n+4}.
It doesn't stay 2^{n+3}.
But the IC stay the same.

Each says

a term = its term number × preceding term + $2^{3+\text{term number}}$

iteration versus recursion

Given

(1) $y_n = 4y_{n-1} - 4y_{n-2}$ for $n \ge 2$ with IC $y_0 = 0$, $y_1 = 2$

you can find y_4 (and similarly, any other y value) by successively finding y_2, y_3, and finally y_4. The process is called *iteration*:

$$y_2 = 4y_1 - 4y_0 = 4\cdot 2 - 4\cdot 0 = 8$$
$$y_3 = 4y_2 - 4y_1 = 4\cdot 8 - 4\cdot 2 = 24$$
$$y_4 = 4y_3 - 4y_2 = 4\cdot 24 - 4\cdot 8 = 64$$

(That's what I did in examples 1 and 2 in the preceding section.)

You can also find y_4 by going backwards until everything is expressed in terms of y_1 and y_0. That process is called *recursion*:

$$\begin{aligned} y_4 &= 4y_3 - 4y_2 \\ &= 4(4y_2 - 4y_1) - 4(4y_1 - 4y_0) \\ &= 4\Big[4(4y_1 - 4y_0) - 4y_1\Big] - 4(4y_1 - 4y_0) \\ &= 4\Big[4(8-0) - 4\cdot 2\Big] - 4(8-0)\ ) = 64 \end{aligned}$$

Here's an iterative program (in Mathematica) to compute values of y_n.

```
y[n_] := Module[{y0 = 0, y1 = 2},
              Do[{y0,y1} = {y1, 4 y1 - 4 y0},{i,2,n}]; y1
              ]
y[4]
64
```

Here's a recursive program (the program calls itself) to compute values of y_n.

```
y[0] = 0;
y[1] = 2;
y[n_]:= 4 y[n-1] - 4 y[n-2]
y[4]
64
```

The recursive program was much faster.

In this section and the next you'll learn how to solve certain kinds of recurrence relations and get a formula for y_n so that you can jump to y_4 (or y_{1000}) immediately.

It will turn out that the solution to (1) is

$$y_n = n2^n$$

Once you have this solution you can get y_4 directly:

$$y_4 = 4 \cdot 2^4 = 64$$

linear recurrence relations with constant coefficients

Let a,b,c,d be fixed constants. Let f_n be a function of n, like n^2 or 2^n or 3.

Recurrence relations of the form

$$ay_{n+1} + by_n = f_n \qquad \text{(1st order)}$$
$$ay_{n+2} + by_{n+1} + cy_n = f_n \qquad \text{(2nd order)}$$
$$ay_{n+3} + by_{n+2} + cy_{n+1} + dy_n = f_n \qquad \text{(3rd order)}$$

and so on are called *linear* recurrence relation with *constant coefficients*.

The function f_n is called the *forcing function*.

If the forcing function is 0 then the rr is called *homogeneous*.

Here are some rr that fit the pattern:

$3y_{n+4} + y_n = 0$ 4-th order, homog

$y_n = 3y_{n-1} - 2y_{n-2} + 3n$ 2nd order, non-homog (forcing function is 3n)

$y_{n+10} = y_{n+5} - y_{n+4}$ 6-th order (not 10-th order), homog

(alias $y_{n+6} - y_{n+1} + y_n = 0$)

Here are some rr that *don't* fit the pattern:

$y_{n+2} - y_{n+1} + ny_n = 0$ (the coeffs aren't all constants -- the coeff of y_n is n)

$y_n + y_{n/2} = 4$ (the subscripts are off)

From now on, *I'll only consider linear recurrence relations with constant coefficients.* The methods for solving given here don't work otherwise.

Until IC are specified, a rr has many solutions (since it can start any way you like). A *general solution* to an n-th order rr is a solution containing n arbitrary constants (to ultimately be determined by n IC). It can be shown that a general solution includes all possible solutions.

superposition rule

If u_n is a solution of $ay_{n+2} + by_{n+1} + cy_n = f_n$
and v_n is a solution of $ay_{n+2} + by_{n+1} + cy_n = g_n$
then
$u_n + v_n$ is a solution of $ay_{n+2} + by_{n+1} + cy_n = f_n + g_n$
ku_n is a solution of $ay_{n+2} + by_{n+1} + cy_n = kf_n$

proof of the $u_n + v_n$ part of the superposition rule

Assume that u_n produces f_n and v_n produces g_n when substituted into $ay_{n+2} + by_{n+1} + cy_n$. Substitute $u_n + v_n$ to see what happens:

$$a(u_{n+2} + v_{n+2}) + b(u_{n+1} + v_{n+1}) + c(u_n + v_n)$$

$$= \underbrace{au_{n+2} + bu_{n+1} + cu_n}_{f_n \text{ by hypothesis}} + \underbrace{av_{n+2} + bv_{n+1} + cv_n}_{g_n \text{ by hypothesis}} = f_n + g_n \quad \text{QED}$$

special case of superposition for homog rr

If u_n is a solution of $ay_{n+2} + by_{n+1} + cy_n = 0$
and v_n is a solution of $ay_{n+2} + by_{n+1} + cy_n = 0$
then
$u_n + v_n$ is a solution of $ay_{n+2} + by_{n+1} + cy_n = 0 + 0$
ku_n is a solution of $ay_{n+2} + by_{n+1} + cy_n = k \cdot 0 = 0$.

In other words, a constant multiple of a solution to a homog rr is also a solution, and the sum of sols to a homog rr is also a solution.

(2) In particular if u_n and v_n are sols to $ay_{n+2} + by_{n+1} + cy_n = 0$ then a *general* solution is $Au_n + Bv_n$.

finding the general sol to a second order homog rr

Look at this example to see the idea behind the solving process:

$$y_{n+2} - 5y_{n+1} + 6y_n = 0 \text{ for } n \ge 2 \text{ with IC } y_0 = 1,\ y_1 = 4.$$

As a guess, try a solution of the form

$$y_n = \lambda^n$$

Substitute into the rr to see what λ, if any, will make it work:

$\lambda^{n+2} - 5\lambda^{n+1} + 6\lambda^n = 0$ (substitute)
$\lambda^2 - 5\lambda + 6 = 0$ (cancel λ^n's)
$(\lambda-3)(\lambda-2) = 0$
$\lambda = 3,\ \lambda = 2$

Now you know that 3^n and 2^n satisfy the rr. By the superposition principle for homogeneous rr, the general solution is

(3) $$y_n = A3^n + B2^n$$

Now plug the IC into (3) and determine A and B.

$$1 = A + B \qquad (\text{set } n = 0,\ y_n = 1)$$
$$4 = 3A + 2B \qquad (\text{set } n = 1,\ y_n = 4)$$

The solution is A = 2, B = −1. So the final answer is

$$y_n = 2 \cdot 3^n - 2^n$$

Here's the rule.

To solve the second order homogeneous recurrence relation

$$ay_{n+2} + by_{n+1} + cy_n = 0 \quad \text{plus IC}$$

or equivalently to solve

$$ay_n + by_{n-1} + cy_{n-2} = 0 \quad \text{plus IC}$$

first get a general solution (a solution with two arbitrary constants) as follows.
Find the roots of the equation

$$a\lambda^2 + b\lambda + c = 0 \quad (\text{called the characteristic equation})$$

If $\lambda = \lambda_1, \lambda_2$ (real unequal roots) then the general solution is

$$y_n = A\,\lambda_1^n + B\,\lambda_2^n$$

If $\lambda = \lambda_1, \lambda_1$ (real equal roots) then the general solution is

$$y_n = A\,\lambda_1^n + B\underline{\underline{n}}\,\lambda_1^n \qquad (\text{step up by } n) \qquad (\text{proof omitted})$$

If the λ's are not real, forget it. I'm skipping this case.
For example,

if $\lambda = -2, 5$ then a general solution is $y_n = A(-2)^n + B\,5^n$

if $\lambda = 2, 2$ then a general solution is $y_n = A2^n + Bn2^n$

if $\lambda = 1, 1$ then a general solution is $y_n = A1^n + Bn1^n = A + Bn$

Once you get the general solution, plug in the IC to determine the two constants.

example 1

To solve

(4) $$y_{n+2} + 3y_{n+1} + 2y_n = 0 \quad \text{with IC} \quad y_1 = 0,\ y_2 = 6$$

first solve the characteristic equation:

$$\lambda^2 + 3\lambda + 2 = 0$$
$$(\lambda + 2)(\lambda + 1) = 0$$
$$\lambda = -2, -1$$

(5) gen $y_n = A(-2)^n + B(-1)^n$ **warning** Don't write -2^n when it should be $(-2)^n$

Now plug in the IC.

$$0 = A\cdot -2 + B\cdot -1$$
$$6 = A(-2)^2 + B(-1)^2$$
$$-2A - B = 0$$
$$4A + B = 6$$

The solution is $A = 3$, $B = -6$. So the final answer is

(6) $$y_n = 3(-2)^n - 6(-1)^n$$

And here's a partial check. By iteration you have

$$y_3 = -3y_2 - 2y_1 = -18$$
$$y_4 = -3y_3 - 2y_2 = 54 - 12 = 42$$
$$y_5 = -3y_4 - 2y_3 = -126 + 36 = -90$$

Using the solution in (6) you can get y_5 immediately:

$$y_5 = 3(-2)^5 - 6(-1)^5 = 3\cdot -32 - 6\cdot -1 = -90$$

solving higher and lower order homogeneous rr

Here's how to generalize the procedure that worked for *second* order homogeneous rr. The 4-th order recurrence relation equation

$$ay_{n+4} + by_{n+3} + cy_{n+2} + dy_{n+1} + ey_n = 0$$

has characteristic equation $a\lambda^4 + b\lambda^3 + c\lambda^2 + d\lambda + e = 0$. If the roots are $\lambda = 2, 3, -4, 6$ then the general sol is

$$y_n = A2^n + B3^n + C(-4)^n + D6^n$$

If you have a 5-th order homog rr and $\lambda = 2, 2, 2, 2, 5$ then a gen sol is

$$y_n = A2^n + Bn2^n + Cn^2 2^n + Dn^3 2^n + E5^n \quad \text{(keep stepping up by n)}$$

If you have a 5-th order homog rr and $\lambda = 1, 1, 1, 2, 2$ then a gen solution is

$$y_n = A1^n + Bn1^n + Cn^2 1^n + D2^n + En2^n$$
$$= A + Bn + Cn^2 \; D2^n + En2^n$$

The same idea works for first order recurrence relations. The rr $ay_{n+1} + by_n = 0$ has characteristic equation $a\lambda + b = 0$. If $\lambda = 4$ then a gen sol is $y_n = A4^n$.

PROBLEMS FOR SECTION 4.2

1. Find the general solution

(a) $y_{n+2} - 3y_{n+1} - 10y_n = 0$ (b) $y_{n+2} + 3y_{n+1} - 4y_n = 0$

(c) $2y_{n+2} + 2y_{n+1} - y_n = 0$ (d) $y_n + 3y_{n-1} - 4y_{n-2} = 0$

2. Solve $y_{n+2} + 2y_{n+1} - 15y_n = 0$ with IC $y_0 = 0$, $y_1 = 1$

3. Given $y_{n+2} - y_{n+1} - 6y_n = 0$ with $y_0 = 1$, $y_1 = 0$

(a) Before doing any solving, find y_3.

(b) Now solve and find a formula for y_n.

(c) Use the formula from part (b) to find y_3 again, as a check.

4. The Fibonacci sequence begins with $y_0 = 0$, $y_1 = 1$ and from then on each term is the sum of the two preceding terms. Find a formula for y_n.

5. Suppose $y_1 = 5$, $y_2 = 7$ and thereafter each term is the average of the two surrounding terms.

(a) Write out some terms and see if you can find a formula for y_n by guessing.

(b) Find a formula for y_n by solving a rr.

6. The λ's for $y_{n+2} - 6y_{n+1} + 9y_n = 0$ are 3 and 3. So 3^n and $n3^n$ are supposed to be solutions.

Check that $n3^n$ really is a solution.

7. Solve $y_n = 4y_{n-1} - 4y_{n-2}$ with IC $y_0 = 0$, $y_1 = 2$.

8. Find a general solution to $y_{n+7} = 2y_{n+5}$

9. If the characteristic equation of a homog rr has the following roots, find a general sol.

(a) −3,4,4 (b) 5,5,5,5,2 (c) 1,1,1,6,−1

10. Go backwards and find a rr with the general sol $y_n = A + Bn + C2^n$.

11. Solve $y_n - 3y_{n-1} + 3y_{n-2} - y_{n-3} = 0$ with $y_1 = 0$, $y_2 = 1$, $y_3 = 0$.

12. Solve by inspection and then solve again (overkill) with the methods of this section.

(a) $y_{n+1} - y_n = 0$ with $y_1 = 4$

(b) $ay_{n+2} + by_{n+1} + cy_n = 0$ with $y_0 = 0$, $y_1 = 0$

13. (a) Find the general solution to the rr $y_n = 5y_{n-1} - 6y_{n-2}$

(b) Check that your solution really satisfies the rr (substitute it into the rr and see that it works).

14. Look at $y_n - 4y_{n+1} + 4y_n = 0$ with IC $y_0 = 7$, $y_1 = 5$.

$\lambda = 2,2$ so you should step and get the general solution $y_n = A2^n + Bn2^n$.

What happens if you mistakenly think the general solution is $y_n = A2^n + B2^n$ (unstepped up) and plug in the IC

SECTION 4.3 SOLVING NONHOMOGENEOUS RECURRENCE RELATIONS

solving a nonhomogeneous rre the idea behind the solving process:
I want to solve

(1) $$y_{n+2} - 5y_{n+1} + 6y_n = 10n + 3 \text{ for } n \ge 2$$

with IC

(2) $$y_0 = 1,\ y_1 = 2$$

It's a non-homog rr because of the $10n + 3$ on the righthand side.
As a guess, try a solution of the form

(3) $$p_n = Cn + D$$

Then

$$p_{n+1} = C(n+1) + D$$
$$p_{n+2} = C(n+2) + D$$

Substitute into (1) to see what values of C and D (if any) make the guess work:

$$C(n+2) + D - 5\Big(C(n+1) + D\Big) + 6(Cn + D) = 10n + 3$$

(4) $$2Cn + 2D - 3C = 10n + 3$$

We want this to be true for all n.
To get it, *match the coeffs of corresponding terms on each side*.
The n coeff on the LHS of (4) is 2C and on the RHS it's 10 so

$$2C = 10$$

The constant term on the LHS of (4) is $2D - 3C$ and on the RHS it's 3 so

$$2D - 3C = 3$$

Solve to get $C = 5$, $D = 9$
Plug into (3) to get the *particular* solution

(5) $$p_n = 5n + 9$$

But p_n doesn't satisfy the IC. Here's how to turn it into a *general* solution, one with the required two arbitrary constants so that you can make it satisfy the IC.
First find the *general homogeneous solution* h_n, the solution to

$$y_{n+2} - 5y_{n+1} + 6y_n = 0$$

The characteristic equation is $\lambda^2 - 5\lambda + 6 = 0$, so $\lambda = 2,3$ and

(6) $$h_n = A\,2^n + B\,3^n$$

The *general* sol y_n to the *non*homog rr in (1) is

$$y_n = h_n + p_n$$
$$= (6) + (5)$$

(7) $$= A\,2^n + B\,3^n + 5n + 9$$

It's general because it contains two arbitrary constants.
And, by superposition, it's still a solution to (1): h_n produces 0 and p_n produces f_n so the sum produces $0 + f_n$ which is f_n.

Now that you have the general solution, plug in the IC.

To get $y_0 = 1$ set $n = 0$, $y_n = 1$ in (7) to get

$$1 = A + B + 9$$

To get $y_1 = 2$ set $n = 1$, $y_n = 2$ in (7) to get

$$2 = 2A + 3B + 5 + 9$$

The solution is $A = -12$, $B = 4$ so the final answer is

$$y_n = -12\cdot 2^n + 4\cdot 3^n + 5n + 9$$

Here's the general rule.

> Here's how to solve
>
> (8A) $$ay_{n+2} + by_{n+1} + cy_n = f_n \quad \text{plus IC.}$$
>
> Let h_n be the *general homog solution* (the sol to $ay_{n+2} + by_{n+1} + cy_n = 0$). Find it with the method of the preceding section.
>
> Let p_n be a *particular nonhomog sol* (a sol, with no constants, to the given nonhomog rr). This section will show you how to find one. Then
>
> (8B) $y_n = h_n + p_n$ is a *general* nonhomog sol
>
> To get the specific solution to (8A), plug the IC into the general solution in (8B) to determine the constants.

finding a particular nonhomog solution for three types of f_n's

Look at

$$ay_{n+2} + by_{n+1} + cy_n = f_n$$

Here's how to find a particular solution p_n in three cases.

The letters A,B,C,D stand for constants.

case 1 f_n is a constant (i.e., there's a plain number on the righthand side)

Try $p_n = A$.

Substitute the trial p_n into the rr to determine the constant A.

case 2 f_n is a polynomial

Suppose $f_n = 7n^3 + 2n$ (a cubic). Try $p_n = An^3 + Bn^2 + Cn + D$ (a cubic not missing any terms even though f_n was missing a few).

Substitute the trial p_n into the rr to determine the constants A, B, C, D.

Similarly if $f_n = 3n^2 + 4n + 1$ (quadratic) then try $p_n = An^2 + Bn + C$ etc.

case 3 f_n is an exponential

Suppose $f_n = 9\cdot 2^n$. Then try $p_n = A2^n$.

Substitute the trial p_n into the rr to determine the constant A.

example 1

Look at

$$y_{n+2} - 5y_{n+1} + 6y_n = 4^n \text{ for } n \geq 3 \text{ with IC } y_1 = 4 \ , \quad y_2 = 0.$$

(a) Find y_5 by iteration.

(b) Solve for y_n.

(c) Find y_5 again using the solution from (b).

solution (a) In general, $y_{n+2} = 5y_{n+1} - 6y_n + 4^n$ so

$y_3 = 5y_2 - 6y_1 + 4^1 = 0 - 24 + 4 = -20$ **warning** The exponent on 4 is 1 not 3.

$y_4 = 5y_3 - 6y_2 + 4^2 = -100 - 0 + 4^2 = -84$

$y_5 = 5y_4 - 6y_3 + 4^3 = -420 + 120 + 64 = -236$

(b) First find h_n.

$$\lambda^2 - 5\lambda + 6 = 0$$
$$(\lambda-2)(\lambda-3) = 0$$
$$\lambda = 2,3$$
$$h_n = B\,2^n + C\,3^n$$

Then look for a particular solution. Try

$$p_n = A\,4^n$$

Substitute into the rr to see what value of A will make it work. You need

$$A\,4^{n+2} - 5A\;4^{n+1} + 6A\,4^n = 4^n$$

Rewrite the left side to display all the 4^n terms:

$$A\,4^2\,4^n \;+ 5A\cdot 4\cdot 4^n + 6A\;4^n = 4^n$$
$$16A\,4^n - 20A\,4^n + 6A\,4^n = 4^n$$
$$2A\,4^n = 4^n$$

The coeff of the 4^n term on the LHS is 2A and the coeff of 4^n term on the RHS is 1 so

$$2A = 1, \quad A = \frac{1}{2}\ .$$

$$p_n = \frac{1}{2}\,4^n$$

(8) $\quad$ gen $y_n = h_n + p_n \;= B\,2^n + C\,3^n + \;\frac{1}{2}\cdot 4^n$

Now plug in the IC.

To get $y_1 = 4$ you need $4 = 2B + 3C + 2$

To get $y_2 = 0$ you need $0 = 4B + 9C + 8$

The solution is $B = 7$, $C = -4$ so the final answer is

(9) $\quad y_n = 7\cdot 2^n - 4\cdot 3^n + \frac{1}{2}\cdot 4^n.$

warning The y_n in (9) is the specific y_n that satisfies the rr and IC. The y_n in (8) is the *general* solution to the rr so you should label it as "gen" to distinguish between the two.

(c) $y_5 = 7\cdot 2^5 - 4\cdot 3^5 + \frac{1}{2}\cdot 4^5 = 224 - 972 + 512 = -236$

warning

1. If the forcing function is n or 5n of −6n try $p_n = An + B$, not just plain An.

Similarly if the forcing function is $3n^2$ or $n^2 + 3$ or $9n^2 + n$, try $p_n = An^2 + Bn + C$, a quadratic *not* missing any terms.

2. Determine the various constants at the appropriate stage.
For a nonhomog rr with IC, first find h_n (containing constants).
Then find p_n (the trial p_n contains constants but they must be immediately determined to get the genuine p_n).
The general solution is $y_n = h_n + p_n$ (contains constants in the h_n part).
Use the IC to determine the constants in the gen sol.

Don't use the IC on h_n alone in the middle of the problem.

stepping up p_n

Look at $ay_{n+2} + by_{n+1} + cy_n = f_n$

Here are some exceptions to the rules in cases 1-3 above

case 1 Suppose $f_n = 6$. Ordinarily you try $p_n = A$.

But if A is already a homog sol (i.e., if one of the λ's was 1) then try $p_n = An$ (step up).

If A and Bn are both homog sols (i.e., λ = 1, 1) then try $p_n = An^2$ (step up some more).

If the rr was *3rd* order and A, Bn, Cn^2 are all homog sols (i.e., λ = 1, 1, 1) try $p_n = An^3$ etc.

case 2 Suppose $f_n = 6n^2 + 3$. Ordinarily you try $p_n = An^2 + Bn + C$.

But if C is a homog sol (i.e., one of the λ's is 1) then try
$p_n = n(An^2 + Bn + C) = An^3 + Bn^2 + Cn$ (step up).

If C and Bn are both homog sols (i.e., λ = 1, 1) try
$p_n = n^2(An^2 + Bn + C) = An^4 + Bn^3 + Cn^2$ etc.

case 3 Suppose $f_n = 9 \cdot 2^n$. Ordinarily you try $p_n = A\,2^n$.

But if 2^n is a homog sol (i.e., one of the λ's is 2) try $p_n = An\,2^n$ (step up).

If 2^n and $n2^n$ are both homog sols (i.e., λ = 2, 2) try $p_n = An^2\,2^n$ etc.

If the rr was *3rd* order and 2^n, $n2^n$, $n^2 2^n$ are all homog sols (i.e., λ = 2, 2, 2) try $p_n = An^3\,2^n$ etc.

example 2

Solve

$$y_{n+2} - 2y_{n+1} + y_n = 2n + 3 \quad \text{with IC } y_0 = 1,\ y_1 = 3$$

First find h_n.

$\lambda^2 - 2\lambda + 1 = 0, \quad \lambda = 1,1$

$h_n = A + Bn$

Ordinarily you would try $p_n = Cn + D$. But D and Cn are already homog solutions so step up and try

$$p_n = n^2\,(Cn + D)$$

Substitute into the rr.

$$(n+2)^2\left[C(n+2) + D\right] - 2(n+1)^2\left[C(n+1) + D\right] + n^2\left[Cn + D\right] = 2n+3$$

When you collect terms, the n^3 terms drop out, the n^2 terms drop out and you're left with

$$6Cn + 6C + 2D = 2n + 3$$

To make this true for all n, make the coeffs of like terms match on each side.
Equate the coeffs of n: $6C = 2$, $C = \frac{1}{3}$

Equate the constant terms: $6C + 2D = 3$, $D = \frac{1}{2}$

So

$$p_n = n^2\left(\tfrac{1}{3}n + \tfrac{1}{2}\right)$$

$$\text{gen } y_n = A + Bn + n^2\left(\tfrac{1}{3}n + \tfrac{1}{2}\right)$$

Now plug in the IC.

To get $y_0 = 1$ you need $1 = A$

To get $y_1 = 3$ you need $3 = A + B + \left(\frac{1}{3} + \frac{1}{2}\right)$

So $A = 1$, $B = \frac{7}{6}$ and the final answer is

$$y_n = 1 + \tfrac{7}{6}n + n^2\left(\tfrac{1}{3}n + \tfrac{1}{2}\right)$$

warning (recurrence relations *not* included in Sections 4.2 and 4.3)

These solving methods are *only* for

(9) $$ay_{n+2} + by_{n+1} + cy_n = f_n \quad \text{where } a,b,c \text{ are constants}$$

and similar equations of higher or lower order. The methods don't work on

(10) $$\boxed{n^3}\, y_{n+2} + \boxed{n}\, y_{n+1} + 6y_n = 8n^2,$$

where the coefficients contain n's instead of being constant.
And they don't work on equations like

(11) $$\boxed{y_n\, y_{n-1}} = y_{n-2} + 3$$

and

(12) $$y_{n+2} - 5\,y_{n+1} + 2\,\boxed{y_n^2} = n$$

which don't have the pattern in (9) at all.

For the equations in (10)–(12), you can find values of y_n by iteration or recursion once you have IC, but, in the context of this course, you can't get a formula for y_n.

PROBLEMS FOR SECTION 4.3

1. Solve (a) $y_{n+2} - y_{n+1} - 2y_n = 1$ for $n \ge 3$ with IC $y_1 = 1$, $y_2 = 3$

(b) $y_{n+2} + 2y_{n+1} - 15y_n = 6n + 10$ for $n \ge 2$ with IC $y_0 = 1$, $y_1 = -\frac{1}{2}$

2.(a) Find a general sol to $y_{n+2} - 3y_{n+1} + y_n = 10\cdot 4^n$.

(b) Rewrite the rr from part (a) so that it involves y_n, y_{n-1} and y_{n-2} and then find the general solution again.

3. Solve $y_{n+2} - y_{n+1} - 6y_n = 18n^2 + 2$ for $n \ge 2$ with $y_0 = -1$, $y_1 = 0$

4. Given $y_n = 2y_{n-1} + 6n$ for $n \ge 2$ with $y_1 = 2$.

(a) Use iteration to find y_4.

(b) Find a formula for y_n.

(c) Use the formula from (b) to find y_4 again as a check.

5. Find a general solution.

(a) $y_{n+1} + 2y_n = 4$ (b) $y_{n+1} + 2y_n = 4^n$

6. Given the following forcing functions and roots of the characteristic equ. What p_n would you try.

	forcing function f_n	λ's
(a)	$n^4 + 2n$	± 2
(b)	$n^4 + 2$	1,1,1,1,3
(c)	$6\cdot 2^n$	2, 6
(d)	$6\cdot 2^n$	3, 6
(e)	3^n	3, 3

7. Solve $2y_{n+1} - y_n = (\frac{1}{2})^n$ for $n \ge 2$ with $y_1 = 2$.

8. Solve $y_{n+2} - 2y_{n+1} + y_n = 1$ for $n \ge 2$ with $y_0 = 1$, $y_1 = \frac{1}{2}$.

9. Let S_n be the sum of the first n squares.

For example $S_5 = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 = 55$.

Find a formula for S_n by solving a recurrence relation.

10. (a) For $y_{n+2} - 3y_{n+1} + 2y_n = 6\cdot 2^n$ you have $h_n = A2^n + B$ so for p_n you should step up and try $p_n = Cn2^n$. What happens if you forget to step up (you dope) and try $p_n = A\,2^n$.

(b) For $2y_{n+2} + 3y_{n+1} + 4y_n = 18n$ you should try $p_n = An \underline{\underline{+ B}}$. What happens if you violate the warning and try $p_n = An$.

11. To get a particular solution to $y_{n+2} - 2y_{n+1} - 3y_n = 5n^2$ you should try $p_n = An^2 + Bn + C$.

Suppose you try $p_n = An$ instead, like this. What's wrong with it.

I need

$A(n+2) - 2A(n+1) - 3An = 5n^2$

$An + 2A - 2An - 2A - 3An = 5n^2$ (multiply out)

$-4An = 5n^2$ (collect terms)

$A = -\frac{5}{4}n$ (solve for A)

So $p_n = -\frac{5}{4}n^2$ QED ???

REVIEW PROBLEMS FOR CHAPTER 4

1. Find a general solution to $y_{n+2} - 9y_n = 56n^2$.

2. Solve $2y_{n+1} + 4y_n = 6\cdot 7^n$ for $n \ge 2$ with IC $y_1 = 5$

3. Find a general solution to $y_{n+2} + 5y_{n+1} - y_n = 6$.

4. Use a difference equation to find a formula for the sum of the first n integers (i.e., a formula for 1 + 2 + 3 + ... + n)

5. Find a gen sol to $y_{n+2} - 9y_n = 5\cdot 3^n$.

6. Suppose you start to solve

$$y_{n+2} - 2y_{n+1} = 0$$

like this:

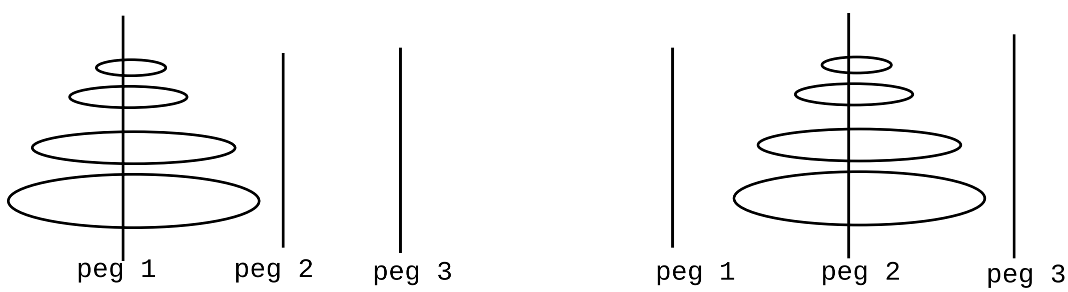
$$\lambda^2 - 2\lambda = 0, \qquad \lambda = 0,\ 2$$

and the gen sol is

$$y_n = A\ 0^n + B2^n = B2^n$$

and you suddenly lost one of the your two constants (which you would need if you were going to satisfy two IC). You've always led a good clean life. How could something like this happen to you and what are you going to do about it.

7.(The tower of Hanoi) The game begins with n rings in increasing size on peg 1. The idea is to transfer them all to peg 2 but never place a larger ring on top of a smaller ring at any stage of the game. Rings may be moved temporarily to peg 3 as they eventually go from peg 1 to peg 2.



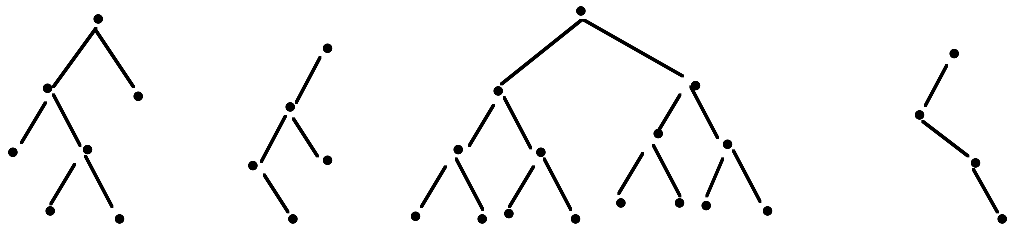
GAME BEGINS GAME ENDS

Let y_n be the minimum number of moves required in a game with n rings (the min number of moves in which the transfer from peg 1 to peg 2 can be accomplished, including moves to and from peg 3).

(a) Write a difference equation for y_n and find IC.

(b) Solve.

8. Here are some examples of binary trees of height 3 (call them 3-trees)



A binary tree begins with one node and then each node has either a left child or a right child or both or neither.

There is one binary tree of height 0, the tree consisting of one vertex and no edges.

There are three binary trees of height 1 (i.e., 1-trees):

Left Child   Right Child   Both Children

Let y_n be the number of binary trees of height n, i.e., the number of n-trees.

(a) Write a difference equation for y_n.

Suggestion: Think of an n-tree as starting with one of the 1-trees followed by an (n−1)-tree.

(b) Use the difference equation from (a) to find y_2 and y_3.

(c) Can you solve the difference equation in the context of this course.

9. A street has n parking spaces along one side (for parallel parking).

A limo takes 2 parking places and a stretch limo takes 3 parking places.

Find a difference equation and IC for the number of ways in which the n spaces can be completely filled with limos and stretch limos.

For example if the street is 5 spaces long then there are two ways:

10. You are going to take out a 3-year $10,000 car loan to be paid off in equal monthly installments of D dollars. The interest rate is 12% a year (1% monthly) The problem is to find D ; i.e., what should the monthly payment be so that the loan is paid off in 3 years.

Suggestion Let y_n be the unpaid balance after n months. Find a difference equation and IC for y_n and solve for y_n. Then use the "end" condition $y_{36} = 0$ to determine D.

CHAPTER 5 FINITE STATE MACHINES

SECTION 5.1 FSMs WITH OUTPUTS

a finite state machine (FSM) with outputs

A FSM (also called a finite automaton) with outputs is an abstract device consisting of a finite number of states (one of which is called the starting state), a finite input alphabet and a finite output alphabet. Initially the FSM is in its starting state. It receives a symbol from its input alphabet, in response emits a symbol from its input alphabet and moves to a next state.

A FSM is considered to have memory (as you'll see in examples) because the current output depends not only on the current input but on the present state of the machine which itself is a summary of the past history of the machine.

The state diagram in Fig 1 shows a FSM with input alphabet 0,1 and output alphabet p,q,r. The states are A (the starting state), B and C. To see how the FSM works let's find the output corresponding to the input string

0 1 1 0 1 0 1.

The leftmost symbol in the string is fed in first. The FSM is initially in its starting state A and when input 0 is received, it produces output p and moves to next state A (i.e., stays in A). After the next input, 1, the FSM produces output q and moves to next state B. The table in Fig 2 lists the successive outputs and next states. All in all, the output string is

p q p r p p q

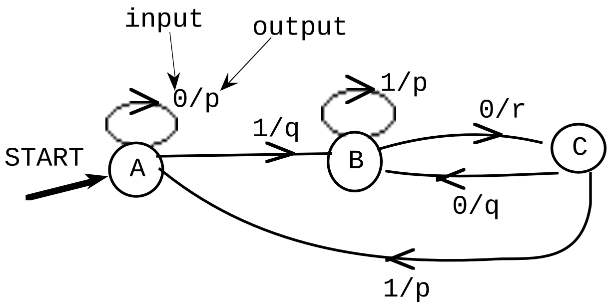


FIG 1

input	0	1	1	0	1	0	1
output	p	q	p	r	p	p	q
next state	A	B	B	C	A	A	B

FIG 2

The state table in Fig 3 contains the same information as Fig 1, but in different form. For example the second line of the table in Fig 3 says that if the FSM is in state B and the input is 0 then the output is r and the machinemoves to next state C; if the input is 1 the output is p and the next state is B.

state	output		next state	
	input 0	input1	input 0	input 1
A	p	q	A	B
B	r	p	C	B
C	q	p	B	A

FIG 3

finite state machines as recognizers

Look at an input string of symbols, called a *word.* The FSM *recognizes* the word if the last output is 1.

It's often necessary to design a FSM to recognize a class of words (e.g., the compiler in a computer must to be able to recognize inputs of a certain form).

example 1

Suppose you want a FSM with input and output symbols 0,1 which recognizes words ending in 101, i.e., which outputs a final 1 if a word ends in 101 and outputs a final 0 otherwise.
Note that if a FSM is to recognize words ending in 101 then it must output a 1 *after every occurrence* of 101 in an input string in case the word suddenly ends there.

Fig 4 shows a sample input string with the corresponding desired outputs.

input	0	0	1	1	1	0	1	0	1	1	1	1	0	1	0	0	0
output	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0

FIG 4

Fig 5 shows the FSM recognizer.

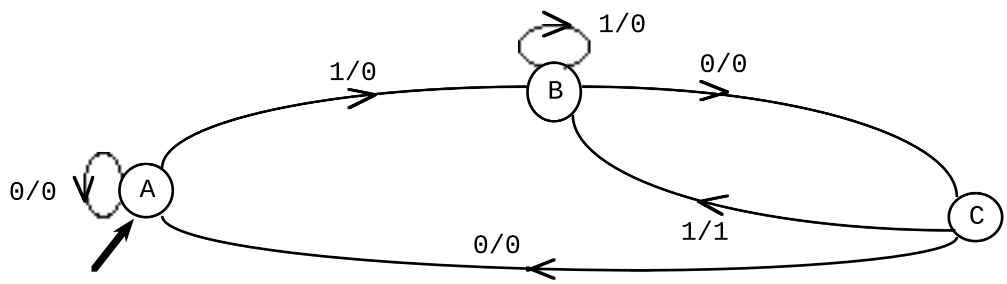


FIG 5 Recognizes words ending in 101

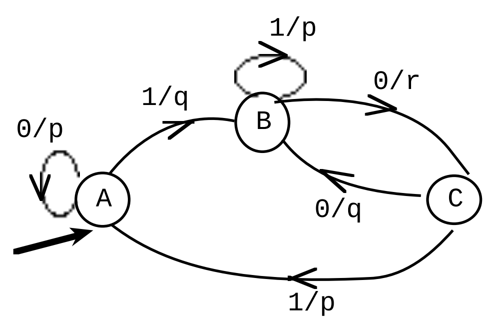
State A "remembers" that (i.e., is reached when) neither of the last two inputs were 1's so that a 101 block isn't in the making yet. The machine won't switch to another state until you do get an input 1, and a 101 block starts to grow.
State B remembers that the last input was 1. The FSM stays in state B until it receives an input 0 and a 101 starts to grow further.
State C remembers that the last two inputs were 10.

If the next input is 1, the output is a 1 to signal that a 101 was completed, and the next state is B since the last input 1 may be the beginning of a new 101.
If the next input is 0 then the output is 0 indicating that a 101 was not achieved, and the FSM sends you back to A to try again.

PROBLEMS FOR SECTION 5.1

1. Fill in the following table for the given FSM

input	0	0	1	1	0	0	0	1	0
output									
next state									



2. Let the input and output symbols be 0,1. Design a FSM to recognize the set of words which
 (a) contain an even number of 1's
(remember that a word with no 1's, i.e., with zero 1's, should be recognized since 0 is an even number)
 (b) contain an odd number of 1's
 (c) contain no 1's
 (d) contain exactly two 1's
 (e) contain at least two 1's
 (f) end in 000

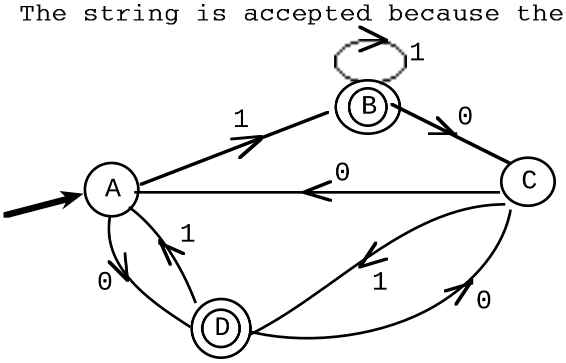
SECTION 5.2 FINITE STATE MACHINES WITH ACCEPTING STATES INSTEAD OF OUTPUTS

definition of a FSM with accepting states

Suppose a FSM is designed with no outputs. Instead, some of the states are called *accepting states*, denoted by a double circle. In Fig 1, states B and D are accepting, A and C are non-accepting. The numbers on the edges are the inputs.

A FSM with outputs (preceding section) recognizes (accepts) an input word if the last output is 1. A FSM with accepting states (this section) recognizes (accepts) a word if the machine is left in an accepting state after the word has been processed.

Fig 1 shows a FSM with accepting states instead of outputs and a table for the input string 1 0 1 1 1.
The string is accepted because the machine ends up in B, an accepting state.



input	1	0	1	1	1
next state	B	C	D	A	$\boxed{B}$

FIG 1

example 1

Fig 2 shows a FSM with accepting states which accepts the set of strings ending in 101

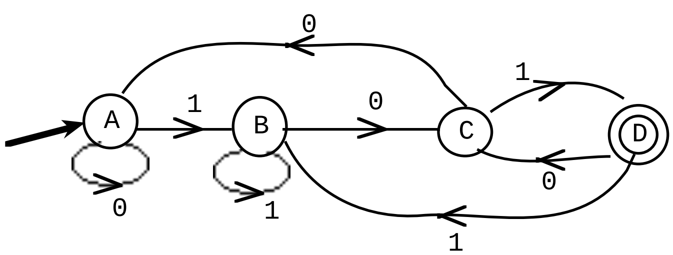
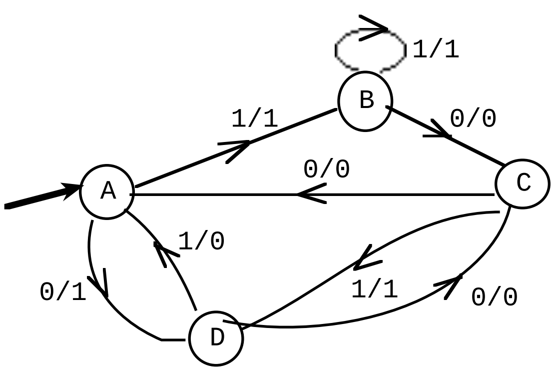


FIG 2

converting a FSM with accepting states to a FSM with outputs

Make the output 1 for each transition into an accepting state and 0 for each transition in a non-accepting state. Fig 3 shows the FSM with accepting states from Fig 1 converted to a FSM with outputs.
The table shows that the string 1 0 1 1 1 is accepted because the last output is 1.



input	1	0	1	1	1
output	1	0	1	0	$\boxed{1}$
next state	B	C	D	A	B

FIG 3 The output version of Fig 1

and vice versa --- converting a FSM with outputs to a FSM with accepting states

I'll illustrate how to do it with the FSM in Fig 4.

When the FSM in Fig 4 enters state A it emits output 0 or 1, depending on the state you're coming from. To make an equivalent FSM with accepting states, replace state A by non-accepting A_0 and accepting state A_1, the former to be entered only as the machine emits 0 and the latter only as the machine emits 1. Similarly B is replaced by non-accepting state B_0 and accepting state B_1 so that the no-output version will have four states, not two. The table in Fig 5 shows how pieces of the original FSM are replaced by new pieces. Fig 6 shows the new FSM with accepting states.

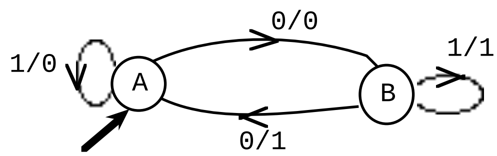


FIG 4 FSM with outputs

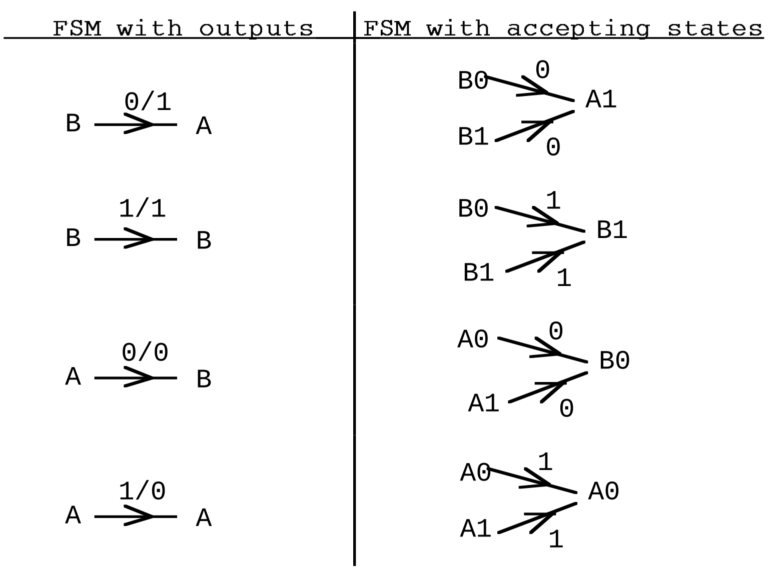
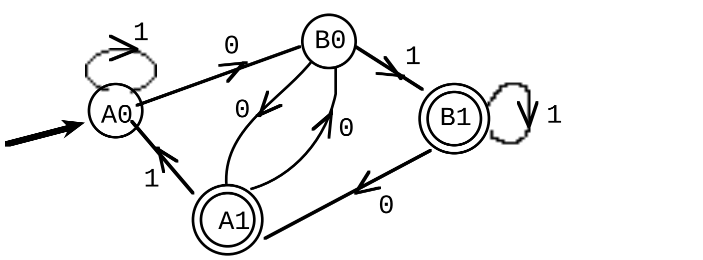


FIG 5



accepting-state version of the FSM in Fig 4

FIG 6

Note that in the new FSM the 0-exits from B0 and B1 must go to the *same* state, (happens to be A1 here). And the 1-exits from B0 and B1 must go to the same state (happens to be B1 here)

In general, if there are transitions into a state Z with outputs of 0 *and* 1 then Z gets split into non-accepting state Z_0 to receive the transitions with output 0 and accepting state Z_1 to receive the transitions that had output 1.

And Z_0 and Z_1 each have the same transitions *out* as the original Z.

recognition capabilities of FSMs with outputs vs. FSMs with accepting states

For every FSM with accepting states there is a FSM with outputs that recognizes the same words. And vice versa.

So *you do not lose or gain recognition capability by switching from FSMs with outputs to FSMs with accepting states.*

footnote Well, actually you do gain recognition capability, in particular the ability to recognize the empty string, denoted λ. But don't worry about it.

The advantage of FSMs with accepting states is that they are easier to prove theorems about and since the two types have the same recognition capabilities, theorems coming later about FSMs with accepting states will also hold for FSMs with outputs.

the empty string

The empty string, i.e., the string with no symbols in it, is denoted λ.

If the starting state of a FSM is an accepting state then the FSM is said to recognize λ.

choosing the starting state in the equivalent no-output FSM

The starting state of the FSM with outputs in Fig 4 is A, so either A_0 or A_1 should be the starting state of the FSM with accepting states in Fig 6

If you use A_1 as the starting state, the FSM with accepting state accepts the empty string λ, since it's in an accepting state before any symbols comes in. But the original FSM in Fig 5 doesn't recognize λ (no FSM with outputs can do that).

So choose A_0 as the starting state rather than A_1.

inventing a new starting state if necessary

I'll find a FSM with accepting states that is equivalent to (recognizes the same strings as) the FSM in Fig 7.

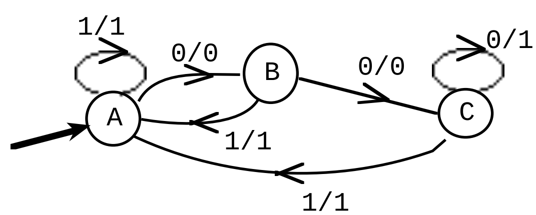


FIG 7

All transitions into A emit output 1 so don't split A into two states; just make A an accepting state in the new version. Similarly all transitions into B emit output 0 so B becomes a non-accepting state (no splitting).

But C is split into a non-accepting state C_0 and an accepting state C_1.

So far you have states A, B, C_0 and C_1 in Fig 8. But you don't want to make A the start since it's an accepting state and you don't want to recognize λ (because the original FSM in Fig 7 does not recognize λ). So add a new state NewStart to serve as the start. *Make NewStart non-accepting, with no transitions in and with the same exits as A.*

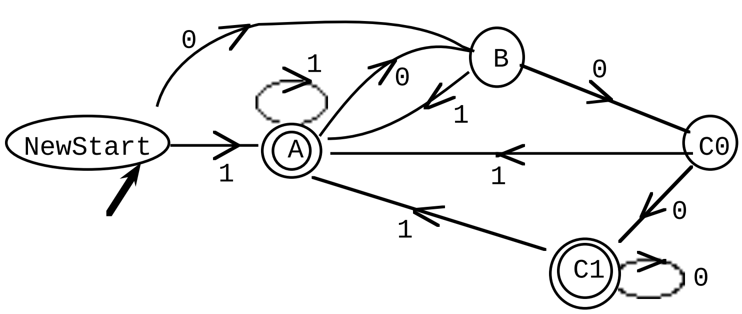


FIG 8

mathematical catechism (you should know the answers to these questions)

question 1

Why bother converting from a FSM with outputs to a FSM with accepting states and vice versa.

answer

The fact that we *can* convert proves that the two collections of FSMs have the same recognition capabilities (give or take recognizing λ).

question 2

Why do we care that the FSMs with outputs and the FSMs with accepting states have the same recognition capabilities.

answer

When we prove a theorem about FSMs with accepting states (what they can and cannot recognize) the same theorem will hold for FSMs with outputs.

PROBLEMS FOR SECTION 5.2

1. Let the input alphabet be 0,1.

Find a FSM with accepting states to recognize the set of strings

(a) containing at least one occurrence of 101

(b) ending in 000

(c) where the number of 1's is a multiple of 3

2. Let the input alphabet be a, b, c.

Find a FSM with accepting states to recognize

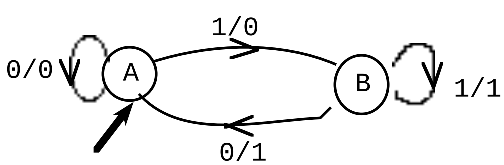
(a) words ending in abc

(b) words containing at least one occurrence of abc

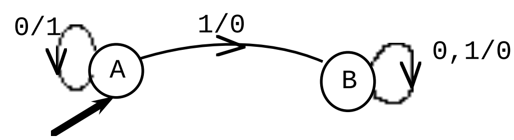
(c) just the word abc

3. For each of the following FSMs use the official method in this section to find an equivalent FSM (i.e., one which recognizes the same strings) with accepting states instead of outputs.

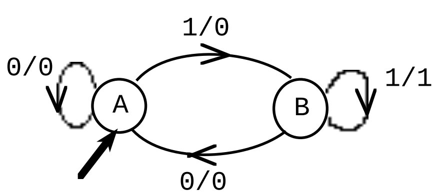
(a)



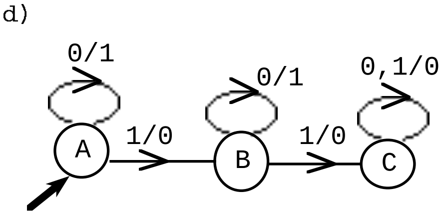
(b)



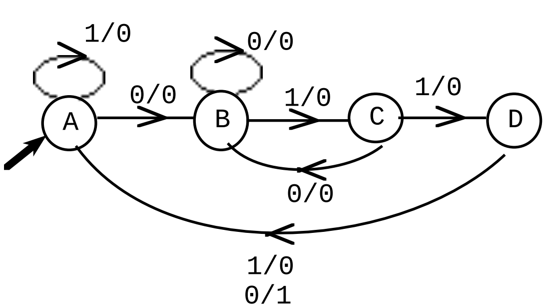
(c)



(d)



(e)



SECTION 5.3 NONDETERMINISTIC FINITE STATE MACHINES

definition of a nondeterministic FSM (assuming the alphabet is 0, 1)

The FSM with accepting states from the last section is a *deterministic finite state machine* (DFSM) meaning that the next state is uniquely determined by the present state and input. From each state there is *one* 0-exit and *one* 1-exit.

In a *nondeterministic finite state machine* (NFSM) there may be more than one 0-exit or 1-exit from a state, i.e., there may be several next states. There may be *no* 0-exit or 1-exit from a state, i.e., the machine may shut down and be in no state at all. For the NFSM in Fig 1, from state A the input 1 allows any of the three next states A, B, C. From state C, any input causes a shutdown.

An input string is recognized (accepted) by a NFSM if the NFSM *may* be left in an accepting state after the word has been processed. You have to look at all the possible tracks to see if any of them reach an accepting state.

For example, try the string 100 in the NFSM in Fig 1. The several possible paths are shown in Fig 2. Since the second path leads to an accepting state, the string is accepted.

warning

Don't say that the NFSM in Fig 1 *might* recognize 100 depending on which track in Fig 2 is followed. The definition says that the NFSM *does* recognize 100 because at least one of the tracks leads to an accepting state.

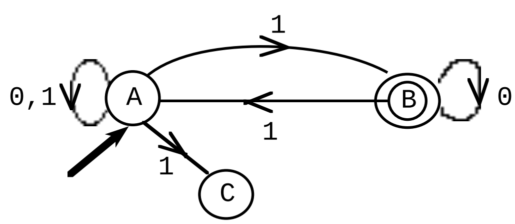


FIG 1 Example of a NFSM

$A \xrightarrow{1} A \xrightarrow{0} A \xrightarrow{0} A$

$A \xrightarrow{1} B \xrightarrow{0} B \xrightarrow{0} B$

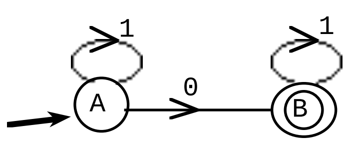
$A \xrightarrow{1} C \xrightarrow{0}$ none

FIG 2 Multiple tracks for input 100

shutdowns

Look at the NFSM in Fig 3 and try input 001. The string is not accepted since the only track leaves the machine in *no* state. (In fact once a string begins with 00 it's not accepted since the machine shuts down after the initial 00.)

Note that from state B with input 0 the next state is "none", *not* B; the machine shuts down and is *not* left in state B.



in		0	0	1
next state	A	B	shutdown	still shutdown

FIG 3

example 1

Fig 4 is a NFSM which recognizes words containing at least one occurrence of cat.

For example, if the input is xcaz*cat*d then there is a track ending in an accepting state (Fig 5) so the string is accepted.

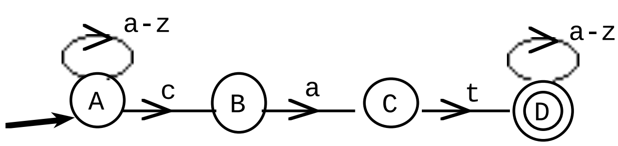


FIG 4

$A \xrightarrow{x} A \xrightarrow{c} A \xrightarrow{a} A \xrightarrow{z} A \xrightarrow{c} B \xrightarrow{a} C \xrightarrow{t} D \xrightarrow{d} D$

FIG 5

the empty string λ

The empty string or null string, i.e., the string containing no symbols, is denoted by λ. Any FSM whose starting state is an accepting state recognizes λ since in that case the FSM is in an accepting state before there is any input at all.

Note that since λ is an empty string, the strings 0λλ1, λ0λ1, 01λλλ are all the same as 01.

NFSMs with λ-moves (λNFSMs)

The NFSM in Fig 6 allows transitions from A to B, B to E, D to B with "input" λ, i.e., with no input. These transitions are called λ-moves.

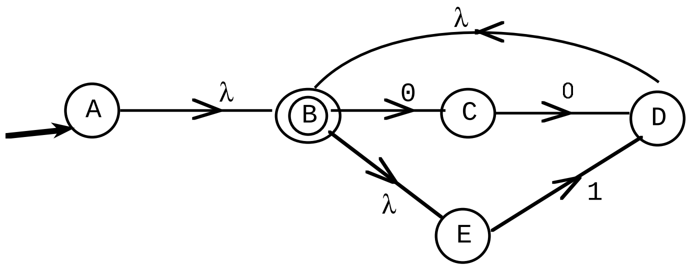


FIG 6

In a NFSM with λ-moves the state Q is said to be λ-*reachable* from state P if there is a sequence of λ-moves leading from P to Q. In Fig 5, E is λ-reachable from A (see the path ABE), B is λ-reachable from D (see the path DB). Furthermore *each state is considered to be λ-reachable from itself*.

In a small FSM like Fig 6 you can tell by inspection which states are λ-reachable from which other states. Here's how to do it in a large FSM. First consider the FSM as a digraph with the states as vertices and the λ-moves as directed edges (ignore other moves). Find the λ-*adjacency matrix* M in which an entry of 1 in row P, col Q indicates a λ-move from P to Q; by convention 1's are inserted along the diagonal to indicate λ-moves from each state to itself. Then use Warshall's algorithm to find the λ-*reachability matrix* M_∞ in which an entry of 1 in row P, col Q indicates a path of λ-moves from P to Q, i.e., indicates that Q is λ-reachable from P.

Here are the λ-adjacency matrix and the λ-reachability matrix for the NFSM in Fig 6.

	A	B	C	D	E
A	1	1	0	0	0
B	0	1	.0	0	1
C	0	0	1	0	0
D	0	1	0	1	0
E	0	0	0	0	1

λ-*adjacency* matrix M for Fig 6

	A	B	C	D	E
A	1	1	0	0	1
B	0	1	0	0	1
C	0	0	1	0	0
D	0	1	0	1	1
E	0	0	0	0	1

λ-*reachability* matrix M_∞ for Fig 6

how to tell whether a NFSM accepts or rejects a string

In a small NFSM you can tell by looking at all the parallel tracks whether or not a string is accepted. Here's an algorithm for deciding automatically.

For a NFSM without λ-moves, keep a record of the *set* of next states (as opposed to the *unique* next state in a DFSM); a string is recognized by the NFSM if you end up with a set of next states which includes an accepting state.

I'll test the string 110 in the NFSM in Fig 7. From the starting state A, the input 1 leads to possible next states A,C (Fig 8). With another input of 1, from A the possible next states are A,C and from C the only possible next state is B; so the set of next states is A,B,C. With the input 0, from A the possible next states are A,B, from B the only possible next state is B and from C there is no next state; so the set of next states is A,B. Since one of them, namely B, is accepting, the string is accepted.

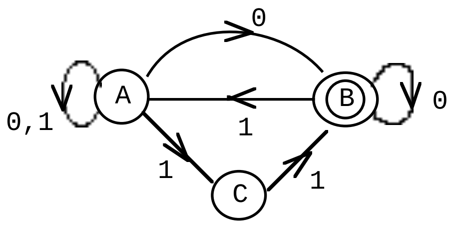


FIG 7

input	1	1	0	
next states	A,C	A,B,C	A, $\boxed{\text{B}}$	accept

FIG 8

For a NFSM with λ-moves there are extra steps because if the NFSM is in say state P then even before the next input appears the NFSM may move to any state λ-reachable from P. So before the first input you should move from the starting state to the set of states that are λ-reachable from the start. And after an input, if the current state is say any of B,C,D then before the next input you should move to the set of states that are λ-reachable from B,C D. This amounts to testing a string as if there were λ's before and after each character in the string .

I'll test 100 in the NFSM in Fig 9.
The starting state is A and states A,B,E are λ-reachable from A so the FSM is considered to be in any of states A,B,E as the first input, 1, is processed (Fig 10). Of these three states only E has a 1-transition, namely to D. Fig 10 continues to show inputs and next states. The string 100 leads to the possible states B,D,E. Since one of these, namely B, is accepting, the string is recognized (in particular see path A B E D B C D B).

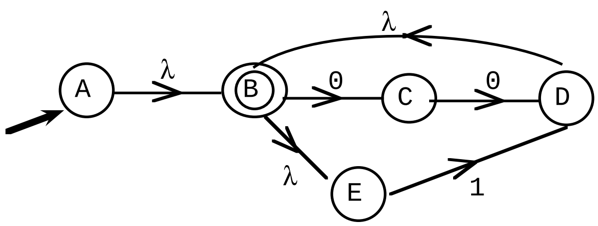


FIG 9

input		λ's	1	λ's	0	λ's	0	λ's	
next states	A	A,B,E	D	B,D,E	C	C	D	$\boxed{\text{B}}$,D,E	accept

FIG 10 The FSM in Fig 9 accepts 100

Let's try the string 110 in the FSM in Fig 9. Fig 11 shows that all tracks lead to the non-accepting state C. So the string is not accepted

input		λ's	1	λ's	1	λ's	0	λ's	
next states	A	A,B,E	D	B,D,E	D	B,D,E	C	C	reject

FIG 11 The FSM in Fig 9 rejects 110

NFSMs versus DFSMs

A DFSM can be physically constructed by engineers so that passage from state to state actually occurs inside the device. A NFSM doesn't exist as a physical device but it is realistic in that its ability to recognize strings can be simulated by a computer program, in particular by the preceding algorithm.

A DFSM can be thought of as a special case of a NFSM, the case where the options of having no next state, more than one next state and λ moves are not exercised. From that point of view it would seem that the class of NFSMs is "larger" than the class of DFSMs. But I'll show:

(1) For every NFSM there is a DFSM which recognizes the same words.

So we *gain no extra recognition capability (and lose none) by expanding from DFSMs to NFSMs*.

The advantage of the NFSMs is that it's easier to prove theorems about NFSMs than about DFSMs and since the two types have the same recognition capabilities, certain theorems coming later about NFSMs also hold for DFSMs.

why (1) holds --- how to convert a NFSM to a DFSM

I'll illustrate the idea by finding a DFSM that recognizes the same strings as the NFSM in Fig 12 (where the starting state is C).

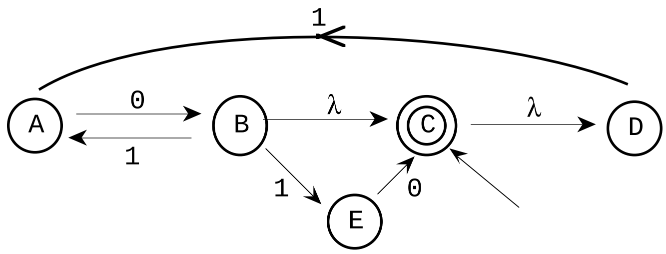


FIG 12

The DFSM will have as its states some or all of the states named

ϕ, A, B, C, D, E, AB, AC, AD, AE, BC, BD, BE, CD, CE, DE, ABC, ABD, ..., ABCD, ..., ABCDE

For convenience you can think that the names correspond to subsets of {A,B,C,D,E}. Here's how to choose the particular states and transitions.

In the NFSM the states that are λ-reachable from the starting state C are C (every state is λ-reachable from itself) and D. So for the new DFSM the starting state will be named CD.

To get the 0-exit from state CD in the new FSM look at 0-exits from C and D in the old NFSM.

C -> none
D -> none

Choose to have a state named ϕ in the DFSM. And use it as the 0-exit from CD.

To find the 1-exit from state CD in the new DFSM look at 1-exits from C and D in the old NFSM.

C -> none
D -> A -> (lambda moves) A

Choose to have a state named A as the next state
(I'm starting to get the DFSM in Fig 13.)

To get the 0-exit from state A in the new DFSM look at 0-exits from A in the old NFSM.

A -> B -> (λ moves) B, C, D

Next state is named BCD.

warning Don't forget to include these λ moves when they exist.
In other words, After A $\xrightarrow{0}$ B, continue from B to all states, including B itself, that are λ-reachable from B, namely B, C, D.

To get the 1-exit from state A in the new DFSM look at 1-exits from A in the old NFSM.

A -> none

Next state is ϕ.

General Rule

(1) (how to get started)
If the starting state in the NFSM is X and the states λ-reachable from X are E, Z then the starting state in the new DFSM is named EXZ.

(2) (how to keep going)
To find the next state from say the state APQ in the new DFSM when the input is 0, look at next states from A, P, Q in the old NFSM when the input is 0.

If say

```
A -> C -> (λ moves) C, X, Y
P -> none
Q -> A, Z -> (λ moves) A, Z, D
```

then the next state in the new DFSM would be named ACDXYZ

If

```
A -> none
P -> none
Q -> none
```

then the next state in the new DFSM would be ϕ.

(3) (transitions from ϕ (if ϕ is a state in the DFSM).
All exits go back to ϕ.

(4) (accepting states)
A state in the new DFSM is called accepting if at least one of the letters in its name was an accepting state in the old NFSM.

Here's the rest of the construction.

To find the 0-exit from BCD, look at 0-exits in the NFSM:

```
B -> none
C -> none
D -> none
```

Next state is ϕ.

To get the 1-exit from BCD, look at 1-exits in the NFSM:

```
B -> E -> (lambda moves) E
C -> none
D-> A -> (lambda moves) A
```

Next state is AE

To get the 0-exit from AE:

```
A -> B -> (lambda moves) B, C, D
E -> C -> (λ moves) C, D
```

Next state is BCD

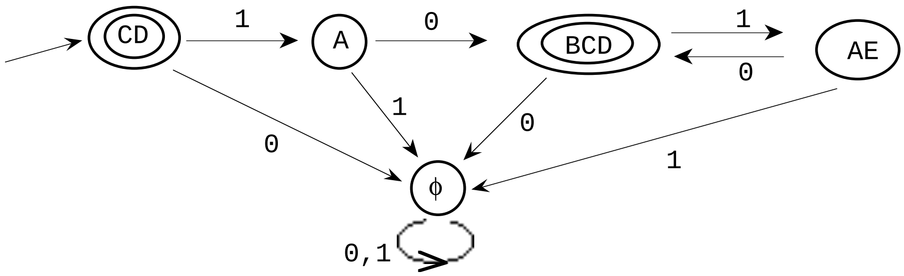
warning Don't forget the λ moves.
If you leave out the λ moves you will mistakenly think that next state is BC.
Messes everything up.

To get the 1-exit from AE:

```
A -> none
E -> none
```

Next state is ϕ.

And finally, C was the only accepting state in the old NFSM so the accepting states in the new DFSM state are the ones whose name contains the letter C (Fig 13).



WARNING

Don't forget the exits from ϕ

And don't forget to say which are the accepting states

FIG 13 DFSM equivalent to Fig 12

To illustrate that the DFSM in Fig 13 recognizes the same strings as the NFSM in Fig 12, I'll test the string 101.

Fig 14A shows the progress of the string through the NFSM in Fig 12. It's rejected because the final set of states, A,E, does not include an accepting state.

Fig 14B shows the progress of the same string through the DFSM in Fig 13. It's rejected because the unique final state, AE, is not an accepting state.

Note that the name of each *unique* next state in the DFSM is an amalgam of the names in the *set* of next states in the NFSM which is why the conversion works.

input		λ's	1	λ's	0	λ's	1	λ's	
next states	C	C, D	A	A	B	B,C,D	A,E,	A,E	reject

FIG 14A

input		1	0	1	
next state	CD	A	BCD	AE	reject

FIG 14B

example 2

Find a DFSM equivalent to (i.e., which recognizes the same strings as) the NFSM in Fig 15.

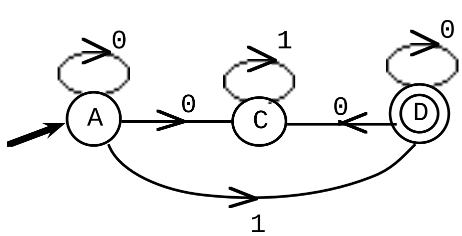


FIG 15

The starting state in the NFSM is A and there are no λ-moves from A so the starting state in the new DFSM will be named A also.

To get the 0-exit from A:

A –> A, C.

Next state is AC

Note: There are no lambda moves in the whole NFSM so I can leave them out.

To get the 1-exit from A:

A –> D.

Next state is D

```
To get the 0-exit from AC:
         A -> A, C
         C -> C
       Next state is AC

To get the 0-exit from D:
          D -> C, D
       Next state is CD

To get the 1-exit from D:
        D -> none
      Next state is φ

To get the 0-exit from CD:
          C -> none
          D -> C, D
       Next state is CD

To get the 1-exit from CD:
           C -> C
           D -> none
        Next state is C

To get the 0-exit from C:
            C -> none
           Next state is φ

To get the 1-exit from C:
            C -> C
           Next state is C
```

The DFSM is in Fig 16. The accepting states are D and CD.

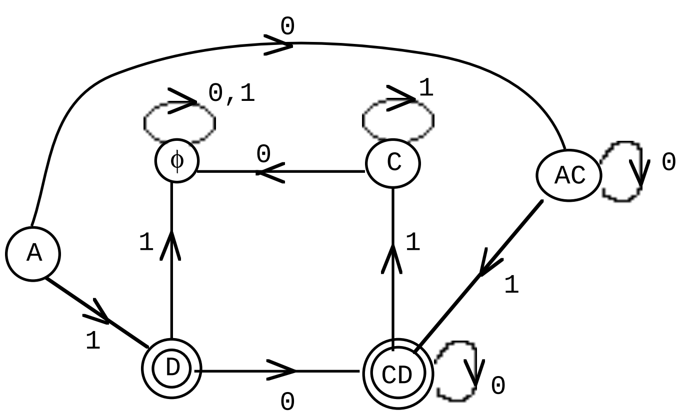


FIG 16

mathematical catechism (you should know the answers to these questions)

question 1

Why bother getting the DFSM recognizer in Fig 16 when it is messier than the NFSM recognizer in Fig 15.

answer The fact that we can always get a DFSM recognizer proves that we don't gain any recognition capability when we allow NFSMs.

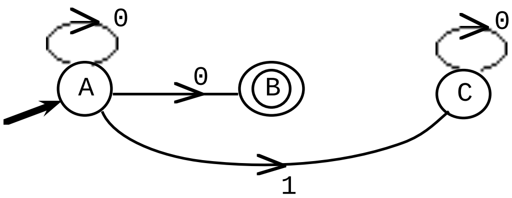
question 2

Why do we care about not gaining capability.

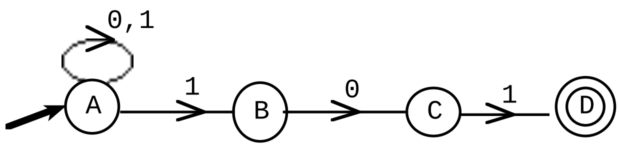
answer When we prove a theorem about NFSMs (what they can and cannot recognize) the same theorem will hold for DFSMs.

PROBLEMS FOR SECTION 5.3

1. Look at the NFSM in the diagram below and decide by inspection whether or not the NFSM recognizes (a) 000 (b) 01



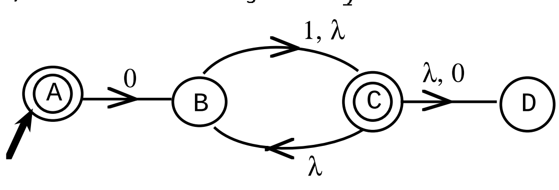
2. What set of strings is accepted by the FSM in the diagram below.



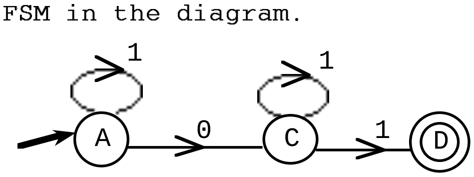
3. Let the alphabet be 0,1. Find FSMs (with accepting states) to recognize the following sets of strings. Try to make them as simple as possible by allowing them to be non-deterministic but do it without using λ moves.
(a) strings ending in 000
(b) strings beginning with 000
(c) just the string 000
(d) strings containing at least one occurrence of 000
(e) strings containing no occurrences of 000
(f) strings containing an even number of 1's
(g) strings with no 1's
(h) strings containing exactly two 1's
(i) strings containing at least two 1's
(j) strings where the number of 1's is a multiple of 3

4. Let the alphabet be a, b, c.
Find FSM recognizers for
(a) the set of strings containing no occurrences of abc
(b) (harder) the set of strings containing exactly one occurrence of abc
Start with a FSM that ends up in an accepting state after the *first* occurrence of abc. And then tack on the FSM from (b) to prevent any more occurrences of abc.
(c) the set of strings containing at most one occurrence of abc.
Suggestion Adjust the FSM from part (b)

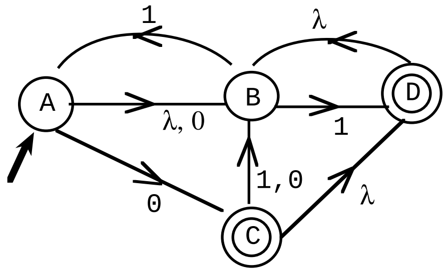
5.(a) Decide by inspection if 011, 0, 00, λ are accepted by the FSM in the diagram.
(b) Find the λ-adjacency matrix and the λ-reachability matrix.



6. Keep track of all next states as in Figs 8, 10, 11 to see if 01110 is accepted by the FSM in the diagram.



7. (a) Keep track of all next states to see if 11 and 00 are accepted by the FSM in the diagram.
   (b) Find the λ-adjacency matrix
   (c) Find the λ-reachability matrix.



8. A NFSM has starting state A, accepting state D, input alphabet a,b and the following a-adjacency matrix (an entry of 1 in row P, col Q indicates an a-transition from P to Q), b-adjacency matrix and λ-adjacency matrix.

a-adjacency

	A	B	C	D
A				
B			1	
C				
D				

b-adjacency

	A	B	C	D
A				
B				
C				1
D		1		

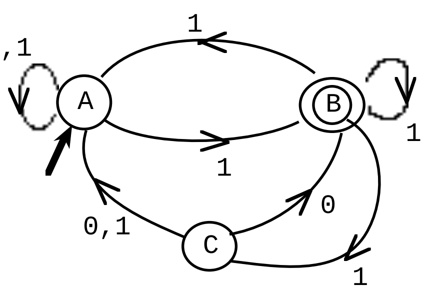
λ-adjacency

	A	B	C	D
A	1	1		1
B		1		
C		1	1	1
D				1

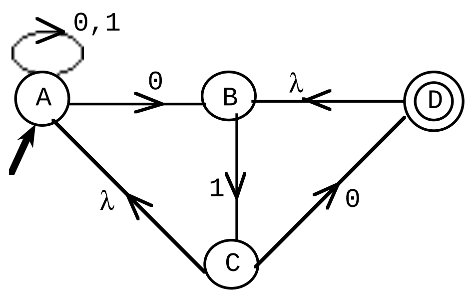
(a) Find the λ-reachability matrix.
(b) Test to see if the string aa is accepted.

9. Find an equivalent DFSM (using the converting algorithm from this section).

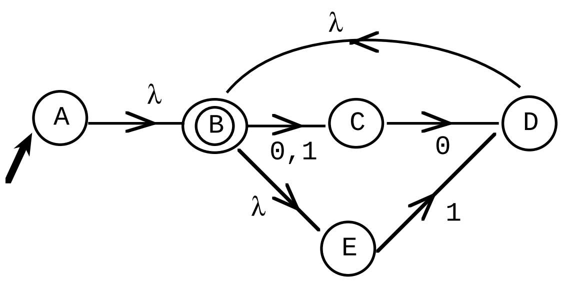
(a)



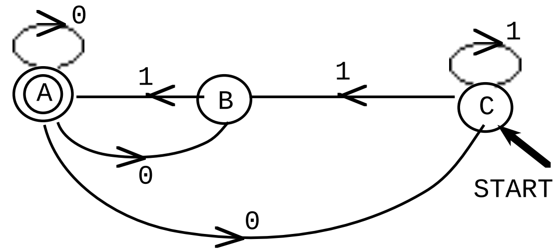
(b)



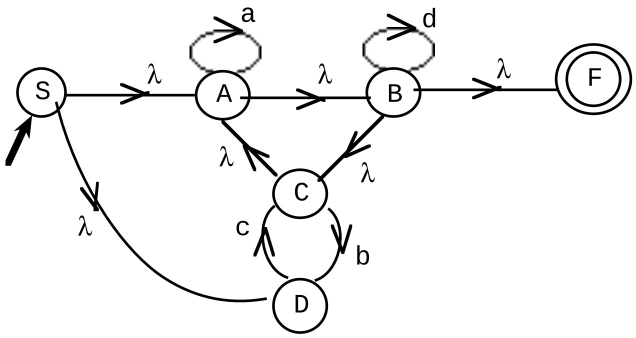
(c)



(d)

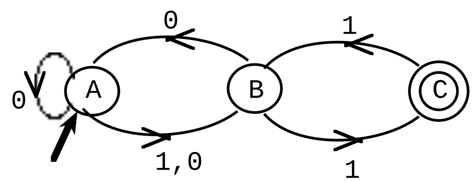


10. If a NFSM contains a λ-cycle then it can be simplified by inspection. See if you can do it for the NFSM in the diagram which has λ-cycle ABCA.



11. Suppose a FSM recognizes a bunch of strings.
The *reverse* of the FSM is a new FSM which recognizes precisely the reverse strings.
For example if the first FSM recognizes cat then the reverse FSM recognizes tac.
Turns out that for every FSM there always is a reverse FSM.

Find the reverse of the FSM in the diagram which conveniently has only one accepting state.



12. Suppose you start with a NFSM and are going to find an equivalent DFSM using the method of this section.
(a) If the NFSM has 10 states what is the maximum number of states in the DFSM.
(b) If the NFSM has 10 states, no parallel tracks but does have shutdowns (i.e., there are never two possible next states but sometimes there is no next state) how many states will the DFSM have.

13. It can be shown that for every FSM with *more than one* accepting state there is an equivalent FSM with just *one* accepting state.
To illustrate the idea let's try to find the one-accepting-state version of the FSM with two accepting states in Fig A below.
(a) I tried to do by merging the two original accepting states (Fig B).
Explain why it doesn't work.
(b) Do it right (pretty easily) (think λ).

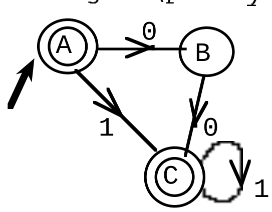


FIG A

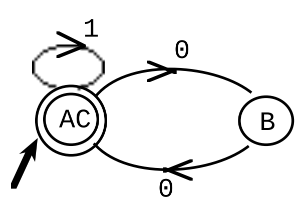


FIG B

SECTION 5.4 REGULAR SETS

concatenation of strings and sets

The concatenation of two *strings* is their juxtaposition; e.g., the concatenation of pq and abc (in that order) is pqabc. The concatenation of the empty string λ with abc (in either order) is just abc, since λ contains no symbols.

If A and B are *sets* of strings then the concatenation AB is the set obtained by concatenating all the strings in A with all the strings in B (in that order).

For example, let A = {a,b,cb} and B = {pq,r}. Then

AB = {apq, ar, bpq, br, cbpq, cbr}
BA = {pqa, ra, pqb, rb, pqcb, rcb}
BB = B^2 = {pqpq, pqr, rr, rpq}

the union of two sets of strings

The union of two sets of strings A and B is denoted A + B.
If A = {a, 00} and B = {pq,c} then A + B = {a, 00, pq, c}

notation I'll write aaaa as a^4 and 01 01 01 as $(01)^3$.

the closure of a set of strings

If A is a set of strings then the (Kleene) closure, denoted A^*, is the set you get by concatenating the strings in A with each other as often as you like in any order you like.

A^* always contains the empty string λ because the definition includes the possibility of concatenating no times

And A^* always contains everything that was in A because the definition allows concatenating just once..

For example, if A = {x} then $A^* = \{\lambda, x, x^2, x^3, \ldots\}$

If A = {a, bc} then words in A^* are formed by stringing together a's and bc's in any order, as many of each as you like. So some strings in A^* are

λ, a, bc, abc, bca, $a^4(bc)^7a^5(bc)^8$, $bca^7(bc)^8$ etc.

notation (ambiguous but convenient)

The set {0} is denoted by 0.

So 0 stands both for the single string 0 and also for the set containing the string 0.

Similarly the set {1} is denoted by 1, the set {b} is denoted by b, etc.

01 can mean just the string 01. Or it can mean the set {01}, the set containing the string 01. It can also be thought of as the concatenation of the string 0 and the string 1.

example 1

0 + 1 is the union of the sets {0} and {1} so it's the set {0, 1}
01 + 0 is the union of {01} and {0} so it's the set {01, 0}
$0^* = \{\lambda, 0, 0^2, 0^3, \ldots\}$.

And especially

$(0 + 1)^* = \{0,1\}^*$ = set of *all* strings of 0's and 1's including the string λ

example 2

To get the members of $(a + b^*)^*$, first get

$$a + b^* = \{a, \lambda, b, b^2, b^3, \ldots\}$$

and then do a lot of concatenating. The result is the set of *all* strings of a's and b's including λ; i.e., $(a + b^*)^*$ is the same as $(a + b)^*$

example 3

$(ab)^* c^* d$ is the set of strings of the form $(ab)^n c^m d$ where $n \geq 0$, $m \geq 0$.
Some of the members of the set are d, abd, $(ab)^5 d$, cd, $c^7 d$, abcd, $(ab)^4 c^{70} d$.

λ versus φ

λ stands for the empty string.
By abuse of notation it also stands for the set containing λ.
φ stands for the empty set.
The sets φ and λ are not the same. One way to see the difference is to look at the different recognizers for them. The FSM in Fig 1 recognizes the string λ, since it's in an accepting state with no input, and recognizes nothing else,. So Fig 1 recognizes the set λ. On the other hand, the FSM in Fig 2 is never in an accepting state (there are no entrances to B), it doesn't accept any strings so the set of strings accepted is φ.

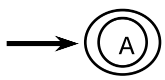


FIG 1 Recognizes the set λ

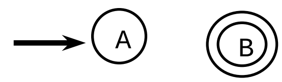


FIG 2 Recognizes the set φ

Furthermore λ and φ act differently when you concatenate or add with them:
$\lambda A = A\lambda = A$
$\phi A = A\phi = \phi$
$A + \phi = A$
$A + \lambda$ might or might not be A (it is A if and only if λ is in A to begin with)
$A - \phi = A$
$A - \lambda$ might or might not be A (it is A iff λ is not in A to begin with)

example 4

To get the members of $(ab^*)^*$, start with $ab^* = \{a, ab, ab^2, ab^3, \ldots\}$ and then concatenate as many times as you like in any order you like. The result is the string λ plus all strings beginning with a. For example, the string $a^5 b^7 a^4 ba$ is in the set since it's of the form $a^4 (ab^7)\ a^3 (ab) a$

The set of strings beginning with a, together with the string λ can also be written as $a(a + b)^* + \lambda$, i.e., $(ab^*)^* = a(a + b)^* + \lambda$

warning

Remember that the closure of any set contains the empty string λ.

regular sets

Let the input alphabet be 0,1 (similarly for any other alphabet).

> If a set of strings can be expressed in terms of the sets 0, 1, λ and a finite number of the operations +, * and concatenation then the set is called regular and the expression itself is a regular expression.
> Furthermore, the set φ is defined as regular.

Equivalently, here's a recursive definition of a regular set.

> (I) (the basic regular sets) The sets 0, 1, λ, φ are regular.
> (II) (building new regular sets from old)
>
> If r_1 and r_2 are regular then so are $r_1 + r_2$, $r_1 r_2$ and r_1^* .

All the sets in examples 1-4 are regular sets.

example 5

The set of strings ending in 10 is regular since it can be written as $(0 + 1)^* 10$

The set of strings containing exactly one 1 is regular since it has the regular expression $0^* 1 0^*$

The set of strings containing at least one 1 is regular because it can be written as $(0 + 1)^* 1 (0 + 1)^*$ [also as $0^* 1 (0 + 1)^*$]

PROBLEMS FOR SECTION 5.4

1. Let A = {ab,c,d) and B = {0,11}. Find A + B, AB, B^2, and find some members of B^*.

2. Let the alphabet be 0,1. Describe in ordinary English the members of the following sets.
(a) 0^* (b) 00^* (c) 0^*0 (d) $(00)^*$ (e) $(0 + 1)^* 111$

(f) $0^* 111$ (g) $0^* 10^* 10^*$ (h) $(0^* 10^* 10^*)^*$ (i) $01 + 111$

3. Two sets P and Q can overlap in any of the five following ways.

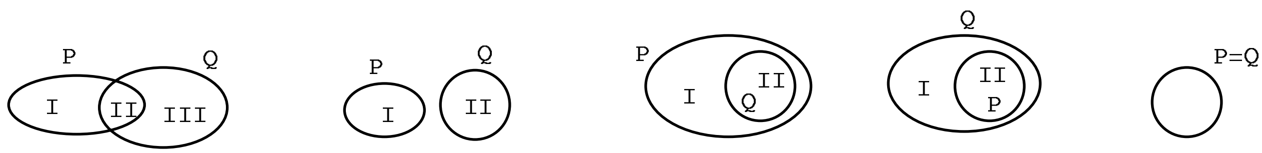


FIG A FIG B FIG C FIG D FIG E

Suspose the alphabet is a,b,c.
Which picture holds for each of the following pairs of P,Q.
And find some strings in subregions I, II, III in each case.

(a) $P = abc(abc)^*$, $Q = a^*(bc)^*$

(b) $P = a^* + (bc)^*$, $Q = (a + bc)^*$

(c) $P = (abc)^*$, $Q = a^* + (bc)^*$

4.(a) Is λ in $a^* b^* c^*$.
(b) Is $01^3 01^3$ in $(1^* 01)^*(11 + 0^*)$.

5. The expression $a^*(a + bb)^*$ can be simplified to $(a + bb)^*$ since the factor a^* in front doesn't contribute anything extra. Simplify the following if possible (by erasing part of the expression).
(a) $(a + bb)^* + bb + b$
(b) $(a + bb)^* bb$
(c) $\left((01)^* + 1\right)^*$

6. Let A and B be arbitrary sets of strings. Simplify if possible,
(a) $(A^*)^*$ (b) $A^* A^*$ (c) $(A^* + B^*)^*$ (d) $AA^* + \lambda$

7. Let A, B, C be arbitrary sets of strings. True or False?
(a) $AA^* = A^* A$
(b) $AB = BA$
(c) $(A^* B^*)^* = (A + B)^*$
(d) $AA^* = A^*$
(e) $A(B + C) = AB + AC$
(f) $1A = A$
(g) $(A + B)^* = A^* + B^*$

8. Let the alphabet be 0,1. Find a regular expression for the set of strings with
(a) no 1's
(b) exactly one 1
(c) least one 1
(d) at most one 1
(e) exactly two 1's
(f) at least two 1's
(g) at most two 1's
(h) an even number of 1's (note that 0 is an even number)
(i) length at least 2 and alternating 0's and 1's such as 10101, 010, 1010, 01
(j) at least one occurrence of 1001

9. Let the alphabet be a,b,c.
Show that the following are regular by finding regular expressions for them.
(a) $\{ab^n a : n \ge 1\}$
(b) $\{a,b,c\}$
(c) set of strings with at least one occurrence of cba
(d) $\{\lambda, a, (bc)^n : n \ge 1\}$
(e) set of strings beginning with a and ending with b
(f) set of strings with second symbol b such as abcca, bb, ab, cbba.

10. (a) The set $(a + bb)^*$ contains λ (every closure contains λ). Find a regular expression for the new set obtained by removing the λ.
You would like to write $(a + bb)^* - \lambda$ but the definition of a regular expression does not allow minus signs. So think of some other way to do it.
(b) The set $a + bc^*$ does not contain λ. Find a regular expression for the new set obtained by including λ.

11. Why is $0^*(0 + 1)\,0^*(0 + 1)\,0^*$ not the set of strings containing at most two 1's and how close does it come.

12. I once asked on an exam for a regular expression for the set of strings in which each 0 is immediately followed by at least one 1, e.g., λ, 1, 11, 011, 1010111 but not 0, 10, 1001.
I got a lot of answers including the following. Which are correct? If one is wrong, explain (to the dope) why it's wrong.

(a) $(1 + 01)^*$

(b) $\left[1^*(01)^*\right]^*$

(c) $\left[1^*\,01\,1^*\right]^*$

(d) $1^* + (011^*)^*$

(e) $1^*(011^*)^*$

(f) $1^*(01)^*1^*$

SECTION 5.5 KLEENE'S THEOREM PART I

Kleene's theorem

Regular sets are the language of FSMs in the following sense. (By FSM I mean DFSMs, in particular, deterministic FSM with accepting states.)

> (I) For every regular set of strings there is a FSM recognizer.
> (II) Conversely, for every FSM, the set of strings it recognizes is a regular set.

In other words, a set of strings is regular if and only if it is the set of strings recognized by some FSM.

footnote
Kleene's theorem almost holds for the FSMs with outputs from Section 5.1. The quibble is that a FSM with outputs cannot recognize the empty string λ, so for example, a FSM with outputs can recognize $(01)(01)^*$ but not $(01)^*$

Kleene's construction method to prove (I)

Although the theorem is about DFSMs, for the proof it's easier to find NFSM recognizers with λ-moves allowed. This is sufficient since I showed in Section 5.3 that for every NFSM with λ-moves there is an equivalent DFSM.

Furthermore, for convenience the proof will use NFSMs which have just one accepting state (which won't be the starting state). This is sufficient since for every FSM with more than one accepting state there is an equivalent NFSM with just one accepting state (Problem 14 in Section 5.3).

In other words I'll show that for every regular set of strings there is a λNFSM with just one accepting state (that isn't the start) that recognizes that set.

I'll show how to construct such a FSM for any regular set by showing two things:

(1) Each of the basic regular sets 0, 1, λ, ϕ has such a FSM recognizer.
(2) Once you have such FSM recognizers for the regular sets r_1 and r_2 you can construct such FSM recognizers for the new regular sets $r_1 + r_2$, $r_1 r_2$ and r_1^* .

The proof of (1) is in Figs 1–4 which show FSMs (with just one accepting state that isn't the starting state) that recognize the sets 0, 1, λ, ϕ.

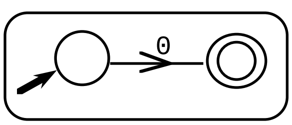


FIG 1
Recognizes 0

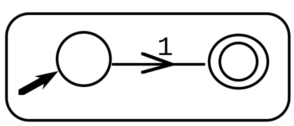


FIG 2
Recognizes 1

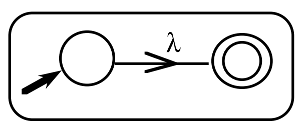


FIG 3
Recognizes λ

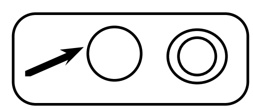


FIG 4
Recognizes ϕ

To prove (2), assume you already have FSMs which recognize r_1 and r_2 (Fig 5).

Recognizes r_1 Recognizes r_2

FIG 5

To make a FSM which recognizes $r_1 + r_2$, change the accepting states in the r_1 and r_2 recognizers to non-accepting and then hook them up in parallel with a new starting state and a new accepting state (Fig 6).

footnote The thing in Fig 6 labeled "r1-recognizer" really is an "r1-recognizer but with its accepting state changed to a non-acc state". But that label was too long to write.

The procedure is illustrated in Fig 7 which shows the recognizers for 0 and 1 from Figs 1 and 2 (with their accepting states made non-accepting) linked in parallel to get a FSM recognizing 0 + 1 (not the most efficient FSM recognizing 0+1 but who cares).

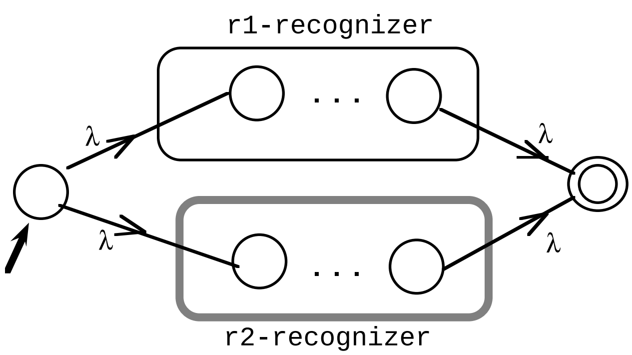


FIG 6 Recognizes $r_1 + r_2$

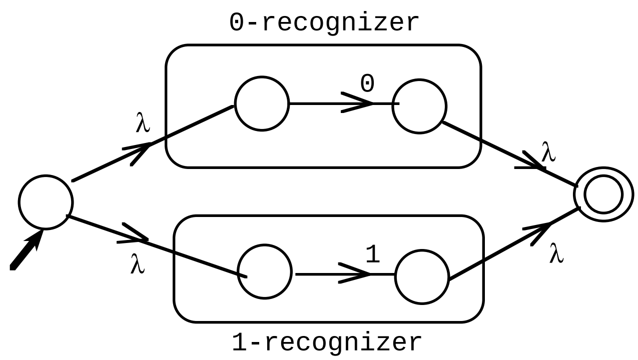


FIG 7 Recognizes 0+1

To find a FSM which recognizes r_1r_2, change the accepting state in the r_1 recognizer to non-accepting and hook up the two recognizers in series (Fig 8).

The procedure is illustrated in Fig 9 which shows the recognizers for 0+1 and 0 from Fig 7 and Fig 1 hooked in series to get a recognizer for (0+1)0.

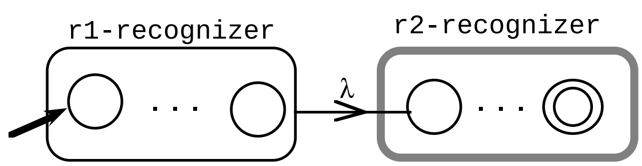


FIG 8 Recognizes r_1r_2

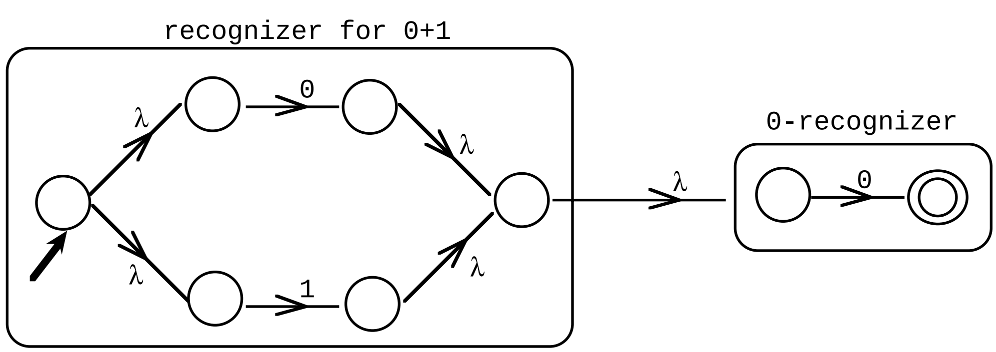


FIG 9 Recognizes (0+1)0

warning

Don't merge the "last" state of the 0+1 recognizer with the "first" state of the 0-recognizer, like Fig 9A.

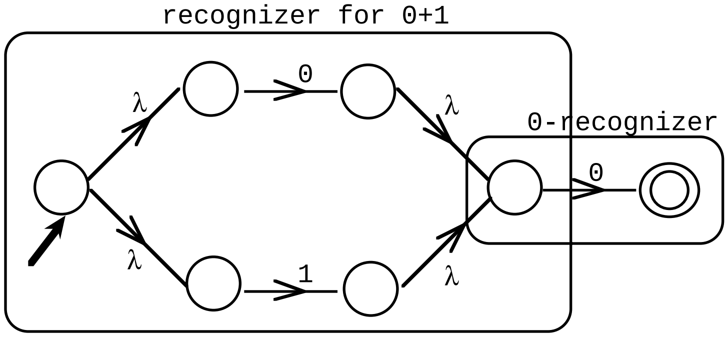


FIG 9A

In this instance, it actually does make a correct recognizer for (0+1)0 *but* i doesn't work in general (see problem #4) so it isn't the way Kleene did it; i.e., it isn't like Fig 8.

To convert the r_1 recognizer to a recognizer for r_1^* (Fig 10) add a new start and a new accepting state connected by a λ-move (so that λ is recognized) and hook up the original machine "circularly" by adding a λ-move from the old accepting state to the old start.

The procedure is illustrated in Fig 11 which shows the recognizer for 0 + 1 adjusted to get a recognizer for $(0+1)^*$.

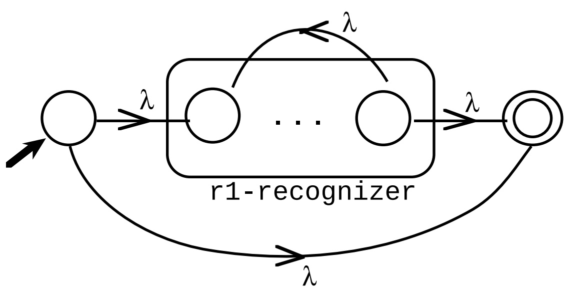


FIG 10 Recognizes r_1^*

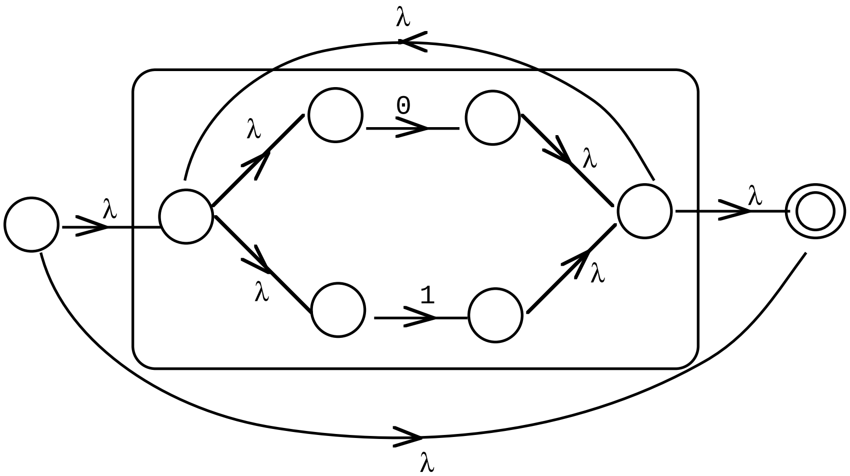


FIG 11 Recognizes (0+1)*

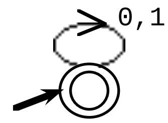


FIG 12

These construction methods are foolproof and are the guarantee that every regular set has a FSM recognizer even though they do not necessarily produce the *simplest* recognizer for a regular set. Fig 12 shows a recognizer for (0+1)* (the set of all binary strings) that is much simpler than the one in Fig 11.

example 1

Figs 13–19 show the Kleene construction of a FSM to recognize (a + b*c)* starting with the FSMs for a,b,c.

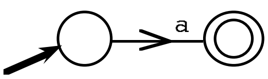


FIG 13 Recognizes a

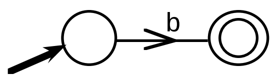


FIG 14 Recognizes b

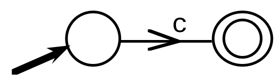


FIG 15 Recognizes c

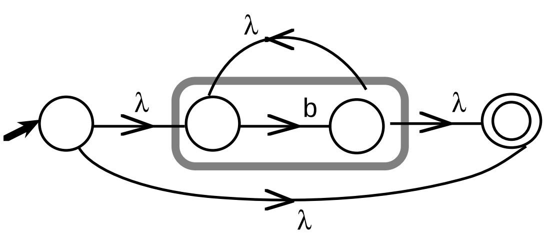


FIG 16 Recognizes b*

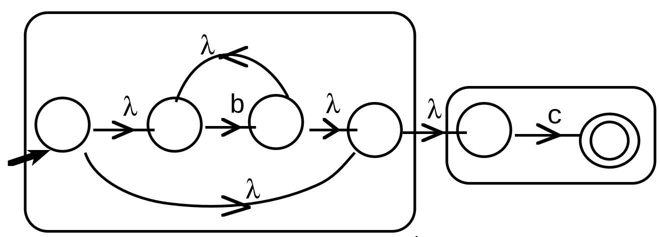


FIG 17 Recognizes b*c

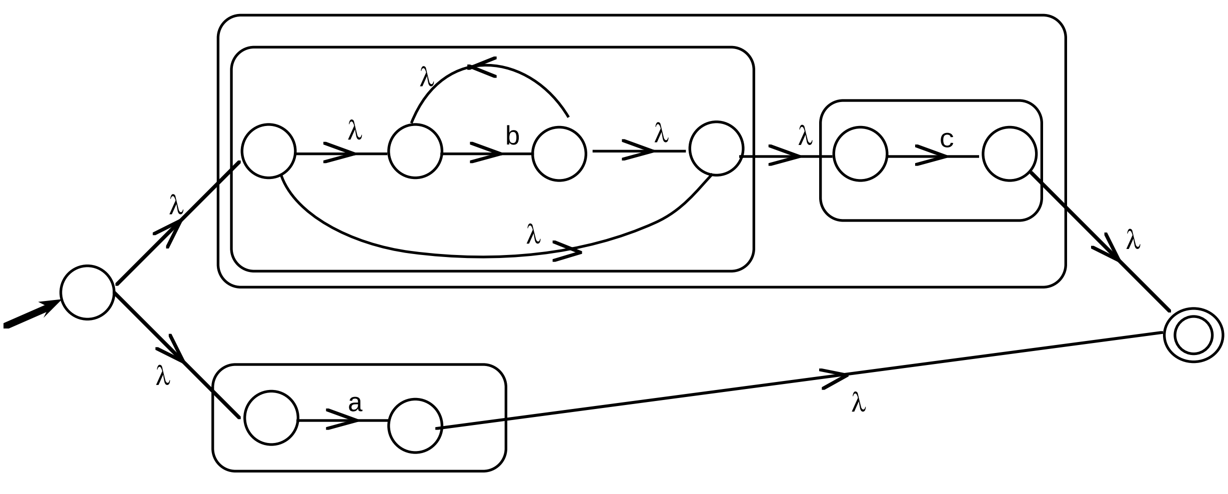


FIG 18 Recognizes a + b*c

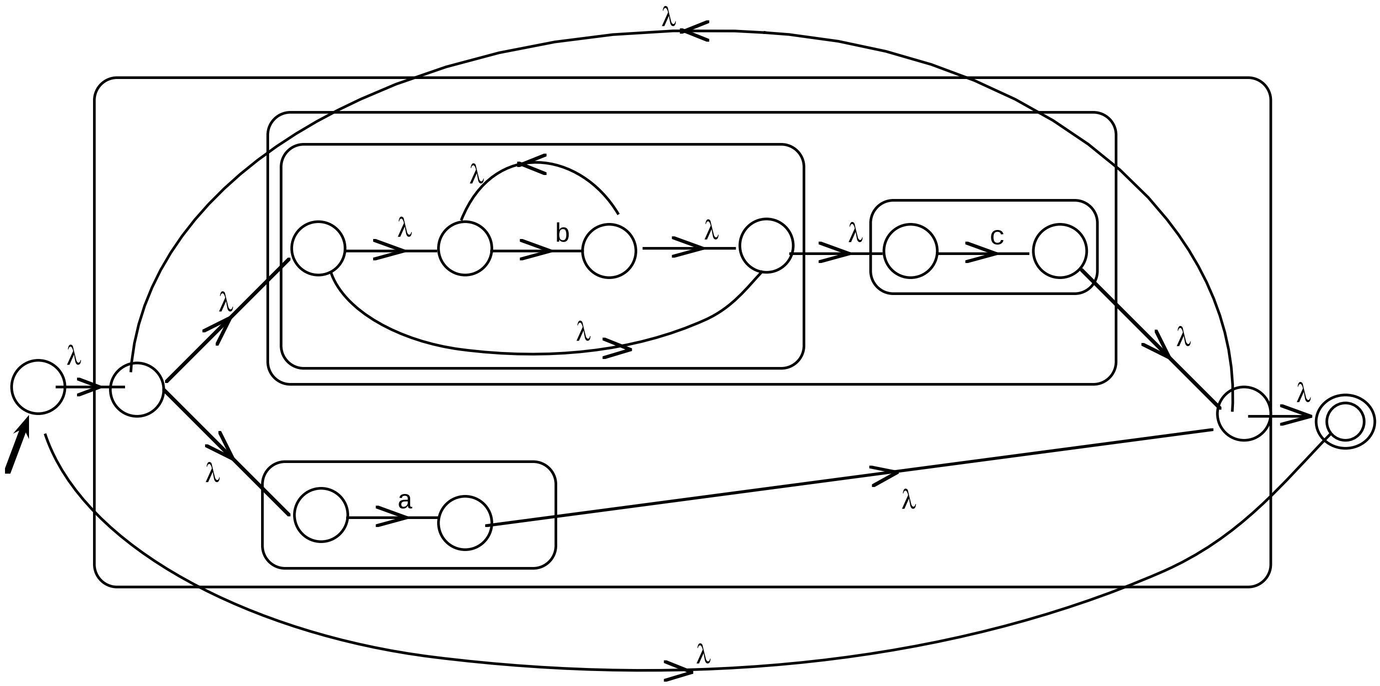


FIG 19 Recognizes $(a + b^*c)^*$

mathematical catechism (you should know the answers to these questions)

question

What good is the Kleene construction method when you end up with a monster FSM.

answer

The Kleene construction is part of the proof that every regular set has a FSM recognizer. Kleene is for showing the existance of a recognizer, not for getting the best recognizer.

PROBLEMS FOR SECTION 5.5

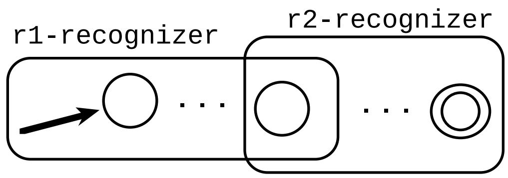
1. Use the Kleene construction method to find a FSM recognizer for each set.
   (a) $(01)^*$ (b) $(01)^*1$ (c) $(01)^* + 1$

2. (a) Find recognizers for the following sets using the Kleene construction method and then find simpler ones by inspection.
   (i) $1^* + 1^*0$ (ii) $01^* + \lambda$

(b) What's the point of the Kleene construction when it produces such an unsimple FSM?

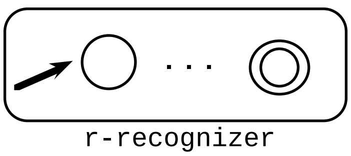
3. Write a regular expression for each set and, by inspection, find a FSM recognizer without λ-moves.
   (a) $\{(01)^n 1^{2m} : n \ge 0,\ m \ge 0\}$
   (b) $\{(01)^n 1^{2m} : n \ge 1,\ m \ge 1\}$
   (c) $\{0^n 1 0^m : n \ge 0,\ m \ge 0\} \cup \{0^k : k \ge 3\}$

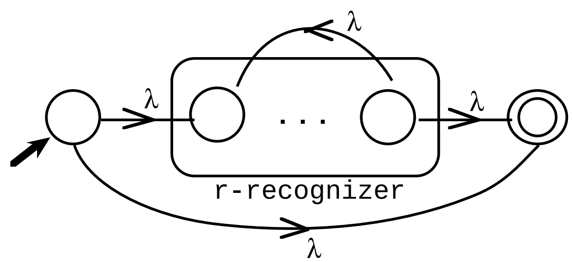
4. Fig 8 showed Kleene's recognizer for $r_1\, r_2$. Here's a proposed streamlined version which merges the accepting state of r_1 with the starting state of r_2 instead of putting in a λ-move from one to the other. Does it work?



Proposed recognizer for $r_1 r_2$

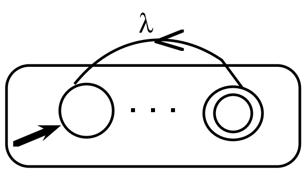
5. On the left below is a hypothetical recognizer for r and on the right is Kleene's recognizer for r* (this is a repeat of Fig 10).



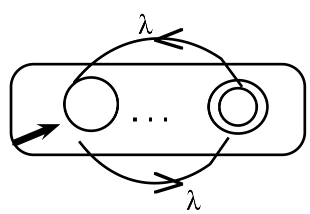


Here are some *other*, sometimes simpler, proposed recognizers for r*.
Do they work? If not, why not?

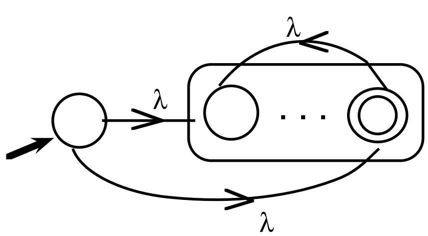
(a)



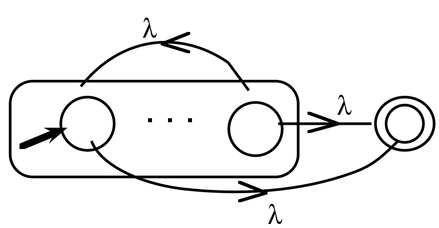
(b)



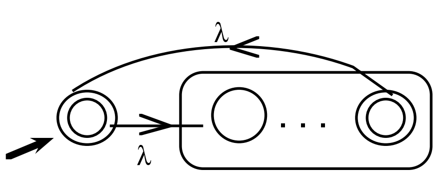
(c)



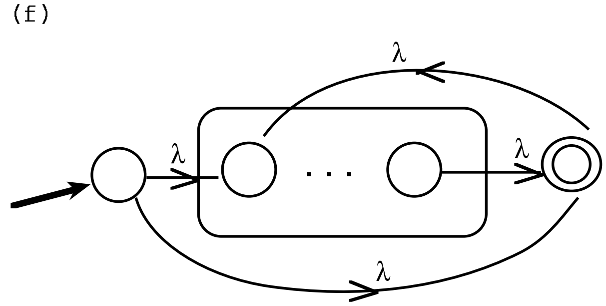
(d)



(e)

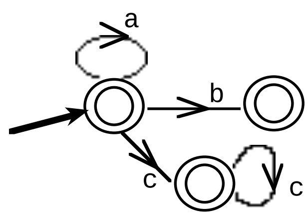


(f)



6. The set $a^*b + c^*$ is a regular set so it must have a FSM recognizer.

(a) What's wrong with the proposed recognizer below

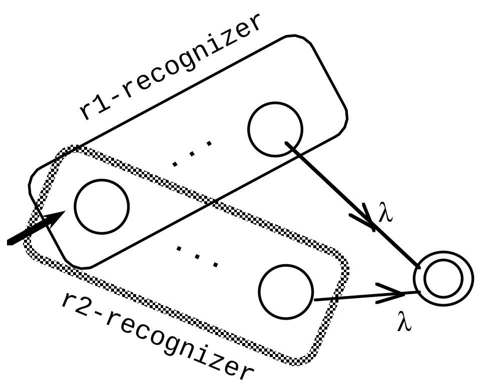


(b) Find a correct FSM recognizer without λ moves and with as few states as possible (by inspection).

7. By inspection, find a FSM recognizer without λ moves and with as few states as possible for

(a) a^*bc^* (b) $(ab + c)^*$

8. Fig 6 showed Kleene's recognizer for $r_1 + r_2$. Here's a proposed streamlined version which merges the two starting states instead of putting in a new start. Show why it doesn't work.



SECTION 5.6 KLEENE'S THEOREM PART II

Part II of Kleene's theorem says that the set recognized by a FSM is regular.

First note that the FSM's in Figs 1 and 2 recognize λ and ϕ respectively. So to keep part II true, with no exceptions, in Section 5.5 we had to (artificially) define λ and ϕ to be regular sets.

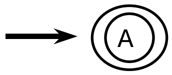


FIG 1 Recognizes the set λ

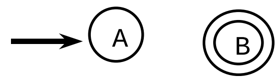


FIG 2 Recognizes the set ϕ

In this section. I'll give a method for actually finding a regular expression for the set recognized by a FSM which proves part II. First some preliminaries.

factoring/combining rule

Let A, B, C, D be sets of strings.

Remember that

AB = set of strings of the form ab where a is in A and b is in B
A + B = A union B = set of strings in either A or B
0A = set of strings of the form 0a where a is a string in A
A1 = set of strings of the form a1 where a is a string in A
$A\lambda = A$
$A\phi = \phi$

Then here are some algebra rules.

$$AB + AC = A(B + C)$$
$$AB + CB = (A + C)B$$
$$(A + B)(C + D) = AC + BC + AD + BD$$

For example, $01A + 0^*A$ can combine to $(01 + 0^*)A$

why the algebra rules work

I'll check on one of them. Let

$$A = \{p,q,r\}, \quad B = \{a,b\}, \quad C = \{v\}$$

Then

AB = {pa, pb, qa, qb, ra, rb}
AC = {pv, qv, rv}
B + C = {a, b, v}

Now you can see that A(B+C) and AB+AC are both {pa, pb, pv, qa, qb, qv, ra, rb, rv}.
So A(B + C) = AB + AC

the P^*Q solving rule

Let P and Q be sets of strings. Suppose P does not contain λ.

(1) The equation $X = PX + Q$ has the unique solution $X = P^*Q$

(2) (special case of (1) where Q is the null set)

The equation $X = PX$ has the unique solution $X = P^*\phi = \phi$

footnote

If P contains λ then $X = P^*Q$ is still a solution to $X = PX + Q$ but it may not be the *only* solution. For example look at the equation

$$X = 0^*X + 1$$

In this case, $P = 0^*$, which *does* contains λ, and $Q = 1$. One solution is

$$X = P^*Q = 0^{**}1 = 0^*1.$$

but it isn't unique. Some other solutions are

$$X = 0^*1 + 0^*11,$$
$$X = 0^*1 + 0^*101$$

and more generally

$$X = 0^*1 + 0^*w$$

where w is any word. But don't worry about it since this won't happen in any of the equations you have to solve; you'll always find that P does *not* contain λ.

half proof of the P*Q rule

I'll show that $X = P^*Q$ *is* a solution of $X = PX + Q$, whether or not P contains λ. The proof that $X = P^*Q$ is the *unique* solution is much harder so I'm leaving it out.

Saying that P^*Q is a solution to the equation $X = PX + Q$ means that if we take P^*Q, multiply in front by (concatenate with) P and add (take union with) Q, you are back where you started; i.e., back to P^*Q. Let's keep track to see.

P^*Q Contains Q strings with P strings in front (as many P strings as you like)
It also contains plain Q strings (using λ from P^*).

PP^*Q Not much different.
Nothing new is created by putting *more* P strings in front.
But something might be lost, namely all the plain Q strings.
If P contains λ then you don't lose the plain Q strings.
If P doesn't contain λ then you do lose the Q strings.

$PP^*Q + Q$ In case the Q strings *were* lost, put them back in.
Now we've got Q strings with P strings in front *and* plain Q strings.

QED

solving the state equations of a FSM to find the regular expression for the set recognized (this proves Kleene's theorem part II)

I'll illustrate the procedure with the FSM in Fig 3.

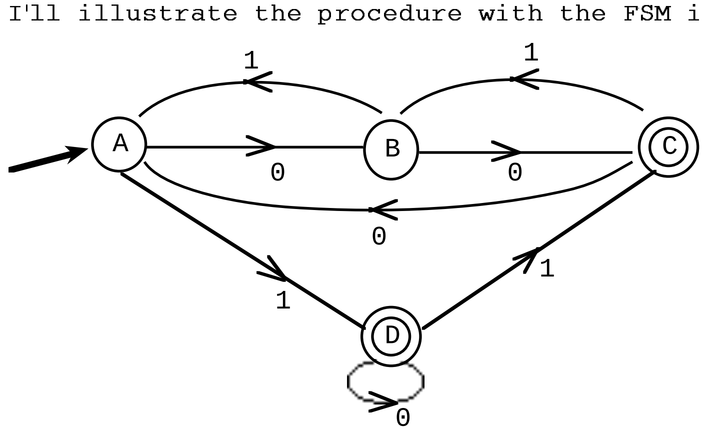


FIG 3

Let A be the set of words which end up in an accepting state starting from state A
Let B be the set of words which end up in an accepting state starting from state B
Let C be the set of words which end up in an accepting state starting from state C
Let D be the set of words which end up in an accepting state starting from state D

I want to find the set A since the FSM in Fig 3 has starting state A. But I'll use the sets B,C,D along the way.

Look at an arbitrary string in the set A, i.e., a string that would end up in an accepting state starting from state A.

If it begins with 0 and you start in state A then the next state is B and the *rest of the word* leads to an accepting state *from B*. So the rest of the word is in the set B and the original string is in 0B.

If it begins with 1 and you start in state A then the next state is D and the rest of the word leads to an accepting state from D. So the rest of the word is in the set D and the original string is in 1D.

Each string in A begins with either 0 or 1 so the strings in the set A are either in 0B or in 1D. So

(3) $\quad A = 0B + 1D$

Similarly

(4) $\quad B = 0C + 1A$

(5) $\quad C = 0A + 1B + \lambda$

(6) $\quad D = 0D + 1C + \lambda$

warning
Don't leave out the λ's in (5) and (6).
They are there because C and D are accepting states.
The λ in (5) reflects the fact that λ leads from C to an accepting state since C itself is accepting.
See problem 11 for what happens if you do leave out λ's

The equations in (3)−(6) are called the *state equations* for the FSM in Fig 3. We want to solve for A since the starting state in Fig 3 is A. To do this, use substitution, factoring and the solving rule in (1). Here is one solution.

I'm going to substitute for B and D in (3) until (3) looks like

$$A = (0{,}1 \text{ stuff})A + 0{,}1 \text{ stuff}$$

and then use the P*Q rule to solve for A.

I'll begin by solving (6) for D using the P*Q rule with $P = 0$, $Q = 1C + \lambda$:

$$D = 0^*(1C + \lambda)$$

Then substitute in (3):

(7) $\quad A = 0B + 10^*(1C + \lambda)$
$\quad\quad = 0B + 10^*1C + 10^* \quad$ (*Note* $10^*\lambda = 10^*$)

Substitute the B value from (4) into (5):

$$C = 0A + 1(0C + 1A) + \lambda$$

Collect terms:

$$C = 10C + 0A + 11A + \lambda$$

And use the P*Q rule to solve for C:

(8) $\quad C = (10)^*(0A + 11A + \lambda)$

Substitute the C value from (8) into (4).

(9) $\quad B = 0(10)^*(0A + 11A + \lambda) + 1A$

Substitute (8) and (9) into (7):

$$A = 0\left[0(10)^*(0A + 11A + \lambda) + 1A\right] + 10^*1\left[(10)^*(0A + 11A + \lambda)\right] + 10^*$$

And collect terms:

$$(10) \qquad A = \left[00(10)^*(0+11) + 01 + 10^*1(10)^*(0+11)\right]A + 00(10)^* + 10^*1(10)^* + 10^*$$

Use the P^*Q rule to solve for A to get the final answer

$$A = \left[00(10)^*(0+11) + 01 + 10^*1(10)^*(0+11)\right]^*\left[00(10)^* + 10^*1(10)^* + 10^*\right]$$

If you do the substitutions in a different order you may get a different expression for A. The expression is not unique.

As a partial check you can trace some strings in A to see that they do reach an accepting state. The beginning of a string in the set A comes from the closure of the set

$$\left\{\ 00(10)^*(0 + 11),\quad 01,\quad 10^*1(10)^*(0 + 11)\ \right\}$$

In other words a string in A begins with blocks of

01's
$00(10)^*(0 + 11)$'s
$10^*1(10)^*(0 + 11)$'s

so here's a typical beginning:

$$(01)^n\ \ 00(10)^m(0 \text{ or } 11)\ \ 10^j1(10)^k\ (0 \text{ or } 11)$$

Then the typical string can continue with a word in

$$00(10)^* \quad \text{or} \quad 10^*1(10)^* \text{ or } 10^*$$

The table in Fig 4 traces the route of this typical string from set A (allowing for three possible string endings) and shows that it *does* end up in an accepting state starting from state A.

(This is at best a partial check; it tries to show that the strings in A *are* recognized but it doesn't check that A includes *all* strings recognized.)

in		$(01)^n$	00	$(10)^m$	0 or 11	10^j	1	$(10)^k$	0 or 11
next	A	A	C	C	A	D	C	C	A

00	$(10)^*$
C	C

10^*1	$(10)^*$
C	C

1	0^*
D	D

FIG 4

warning

1. Don't forget that the state equation corresponding to an *accepting* state has a λ term
2. Solve for the right letter, the one that was the starting state in the FSM.

difference of two sets

If A and B are sets then A − B is the set of things in A that are not also in B. Fig 5 shows some pictures of A − B in various circumstances.

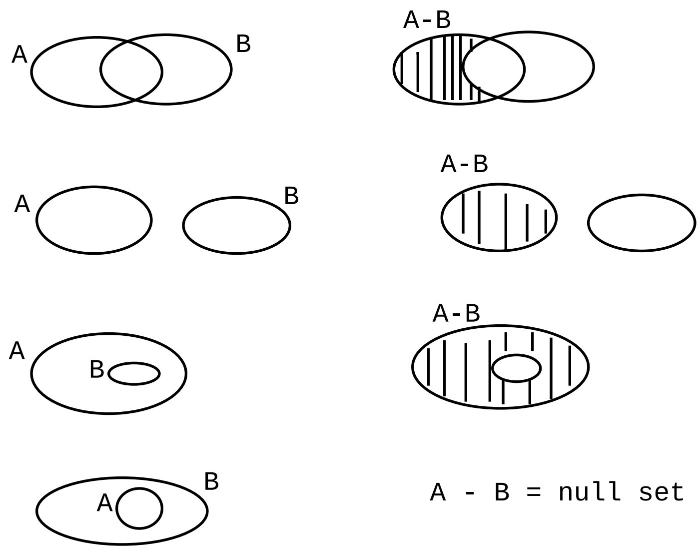


FIG 5

corollary of Kleene's theorem: unions, intersections, complements, differences of regular sets

Let r_1 and r_2 be regular sets of strings using some fixed alphabet. Then the following sets are also regular:

$$r_1 \cup r_2, \quad \overline{r_1}, \quad r_1 \cap r_2, \quad r_1 - r_2$$

proof that the union of regular sets is regular

$r_1 \cup r_2$ is the same as $r_1 + r_2$ and by definition of regular sets if r_1 and r_2 are regular then so is $r_1 + r_2$

proof that the complement of regular sets is regular

By Kleene's theorem part I there is a FSM recognizing the regular set r_1. Find a *deterministic* version (can always do that by Section 5.3) so that every string follows exactly one track; the words in r_1 lead to an accepting state and the words in $\overline{r}_1$ lead to a non-accepting state. Change every accepting state to non-accepting and every non-accepting state to accepting. The new FSM recognize $\overline{r}_1$. Therefore by Kleene's theorem part II, $\overline{r}_1$ is regular.

As an example, Fig 6 shows a DFSM recognizing 0 and the new machine recognizing everything (including λ) except 0.

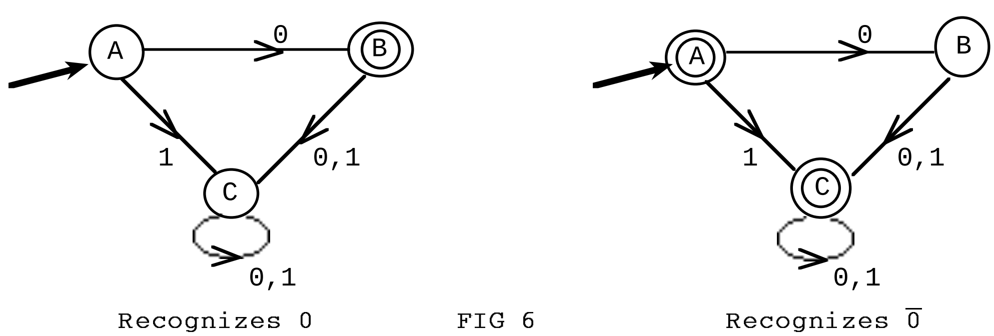


FIG 6

proof that the intersection of regular sets is regular

Write the interesection in terms of complements and unions:

$$\overline{r_1 \cap r_2} = \overline{r_1} \cup \overline{r_2} \qquad \text{(DeMorgan's law of set theory)}$$

$$r_1 \cap r_2 = \overline{\overline{r_1 \cap r_2}} = \overline{\overline{r_1} \cup \overline{r_2}}$$

But as already proved about complements and unions, if r_1 and r_2 are regular then so are $\overline{r_1}$ and $\overline{r_2}$ and so is $\overline{r_1} \cup \overline{r_2}$ and so is $\overline{\overline{r_1} \cup \overline{r_2}}$. So $r_1 \cap r_2$ is regular.

proof that the difference of regular sets is regular

See problem 7.

summary of how to show that a set is regular

You now have three ways to show that a set, r, of strings is regular.

1. Find a reg expression for r (then the set is regular by definition).
2. Find a FSM recognizer for r (then the set is reg by Kleene part II).
3. Show that r is the complement, union, intersection or difference of regular sets (then r is regular by the corollary above).

How do you know which one to use? There's no absolute rule.

You might want to use method 3 if the strings in r must have *two* patterns (then r is an intersection) or if the strings in r must *not* have a certain pattern (then r is a complement) or if the strings in r have one pattern but must avoid another (then r is a difference) or if the strings in r have a choice of patterns (then r is a union/sum).

example 1

Let

A = set of strings ending in 10 *and* containing at least two 1's
B = set of strings containing at least two 1's but *not* ending in 10
C = set of strings *not* ending in 10
D = set of strings ending in 10 *or* containing at least two 1's

Show that they are regular sets.

solution

The strings in A must have two patterns. So I'll think of A as an intersection. In particular let

E = set of strings ending in 10
F = set of strings containing at least two 1's

Then $A = E \cap F$.
E is regular since $E = (0+1)^*10$.
Alternatively, E is regular since Fig 7 shows a FSM recognizer for E.
And F is regular since $F = (0 + 1)^*1(0 + 1)^*1(0 + 1)^*$.
Alternatively, F is regular since Fig 8 is a FSM recognizer for F.
So A is regular since the intersection of regular sets is reg.

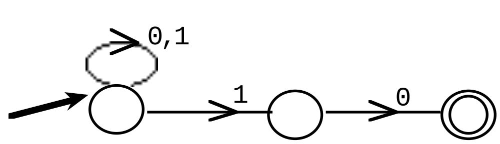


FIG 7

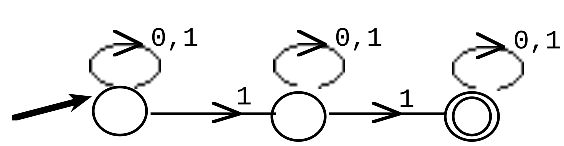


FIG 8

The strings in B must have one pattern but avoid a second. With the E and F as above, B = F − E.
E and F are regular sets, so B is also reg.

The strings in C must avoid a pattern. In particular, C = $\overline{E}$. We already know that E is regular, so C is also reg.

The strings in D have a choice of patterns. In particular, D = E + F (the union of E and F). E and F are regular, so D is also reg. QED

clarification

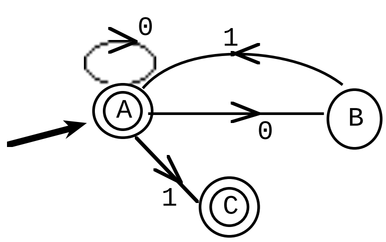
A regular expression can't contain a minus sign or a complement sign. The set $\overline{(0+1)^*10}$ is regular because $(0+1)^*10$ is reg and the complement of a regular set is regular. But the expression $\overline{(0+1)^*10}$ itself is not a regular expression for the set.

Similarly, the set $(01)^* - 0101$ is a regular set because the difference of two regular sets is regular but the expression $(01)^* - 011$ itself is not a regular expression for the set.

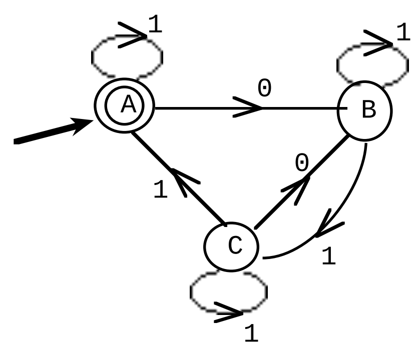
PROBLEMS FOR SECTION 5.6

1. For the following FSMs it may be possible to find a regular expression by inspection for the set of strings recognized. But use state equations for practice (and why not try checking your answer a little).

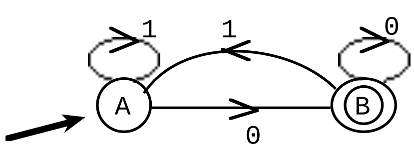
(a)



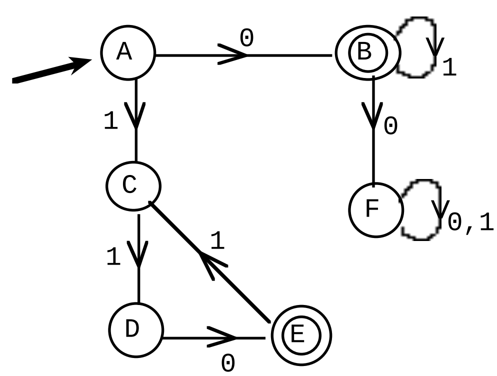
(b)



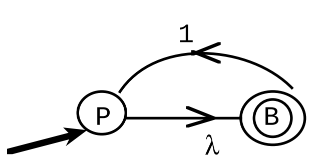
(c)



(d)



(e)



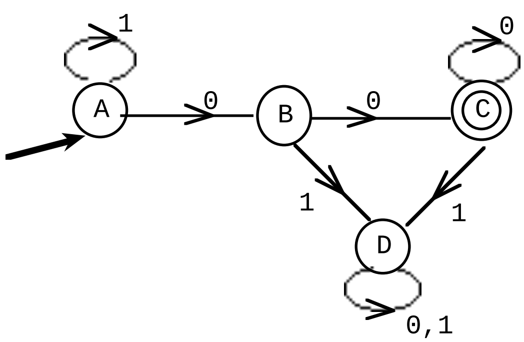
2. Look at the state equations

$$A = 0A + 0B + 1C + \lambda$$
$$B = 1A + 0B$$
$$C = 0C + 1B$$

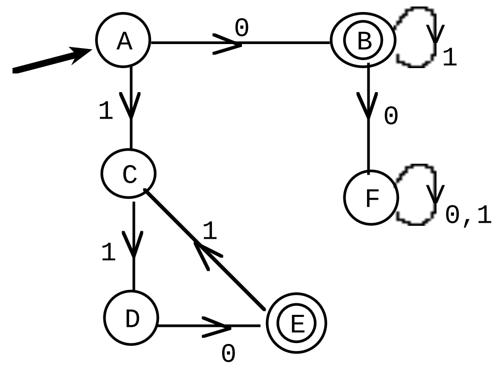
Can you tell which is the starting state? which are the accepting states ?

3. Solving state equations is foolproof and is the guarantee that the set recognized by a FSM is regular but sometimes you can get a regular expression for the set recognized by a FSM by inspection. Try it for these FSMs.

(a)

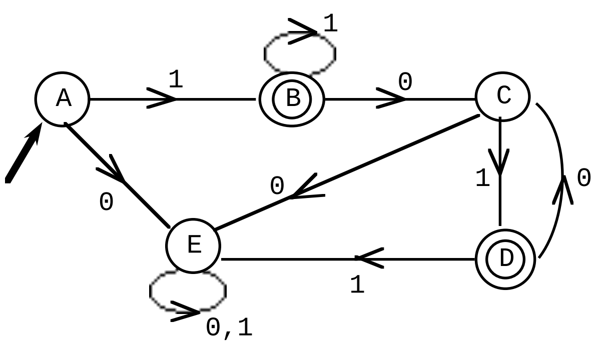


(b)

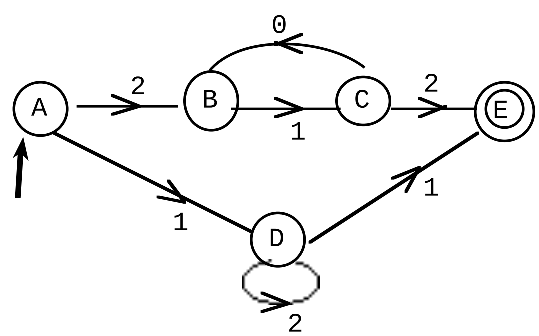


(c) Same as (b) but make the starting state accepting.

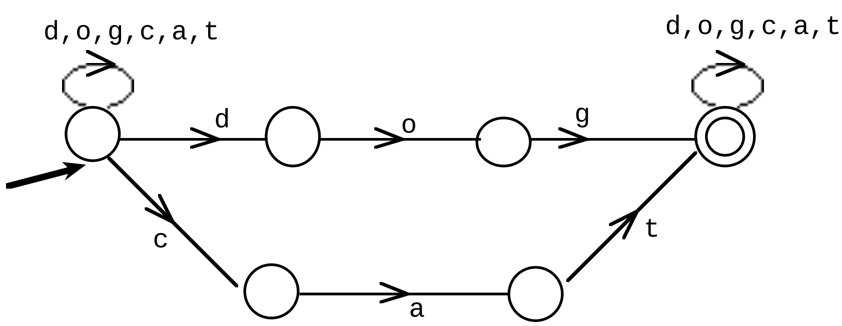
(d)



(e) (the alphabet is 0,1,2)



4. By inspection, find a regular expression for the set recognized by the FSM below and describe the set in words (the alphabet is d,o,g,c,a,t).



5. Let the alphabet be a,b,c,d. Show that the following are regular sets.

r_1 = set of strings with at least one occurrence of abc

r_2 = set of strings with at least two occurrences of abc

r_3 = set of strings with at least three occurrences of abc

r_4 = set of strings with no occurrences of abc

r_5 = set of strings with exactly one occurrence of abc

r_6 = set of strings with exactly two occurrences of abc

6. Show that the following are regular sets.

(a) r = set of strings containing 001 and containing an even number of 1's
(b) s = set of strings containing 001 and not containing 0000
(c) t = set of strings not ending in 0010

7. I already proved that complements and intersections of reg sets are reg. Now give prove that the difference of reg sets is reg by writing the difference in terms of intersections and complements.

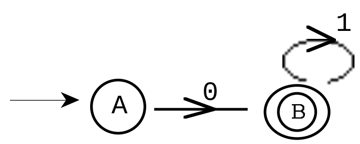
8. Let r be a set of strings and let s be the set of reversed strings. For example if adc is in r then cda is in s. Show that if r is regular then s is regular.

9. (a) What theorem does the Kleene construction stuff in the preceding section prove.
(b) What theorem does the state equation stuff in this section prove.

10. Let r be the set of strings containing *no* occurrences of 1101.

Then $\overline{r}$ is the set of strings containing at least one occurrence of 1101 and it is regular because it has the reg expression $(0 + 1)^*\ 1101\ (0 + 1)^*$.
So r is regular. But I don't see how to get a regular expression for r by inspection.
Outline some steps that could be implemented to get it (don't actually do the implementing) (too tedious).

11. By inspection, the set of strings recognized by the FSM in the diagram below is 01^* but we're going to look at state equations anyway for practice.



The correct state equations are

(1) $A = 0B$
(2) $B = 1B + \lambda$

Suppose you make a mistake and leave out the λ in the second equation so that your (incorrect) state equations are

(3) $A = 0B$
(4) $B = 1B$

(a) Predict (just think!) the answer you should get if you solved the wrong state equations in (3) and (4) for A.
(b) Now solve the wrong equations and see what you actually do get.
(c) Solve the *correct* state equations in (1) and (2)

SECTION 5.7 SOME NON-REGULAR SETS (SETS THAT NO FSM CAN RECOGNIZE)

example 1

Let A be the set of strings of the form $0^n 1^n$, $n \geq 0$. In other words,

$$A = \{\lambda, 01, 0^2 1^2, 0^3 1^3, \ldots\},$$

the set of strings consisting of any number of 0's (including none of them) followed by an equal number of 1's. I'll show that the set is not regular.

This will be a proof by contradiction)

Suppose the set is regular. (I'm hoping to get a contradiction.)

Then it has a FSM recognizer (by Kleene part I).

Let N be the number of states in the FSM.

Look at the string $0^N 1^N$. The string is in A so it is recognized by the FSM. So there is a track (Fig 1) leading to an accepting state. In the 0^N part of the table there are N+1 next-state boxes, one for each of the 0's plus a starting state. But there are only N states. More boxes than states. So the boxes in the 0 part of the track must have a repeated state, say Z. So the track includes a loop (Fig 2).

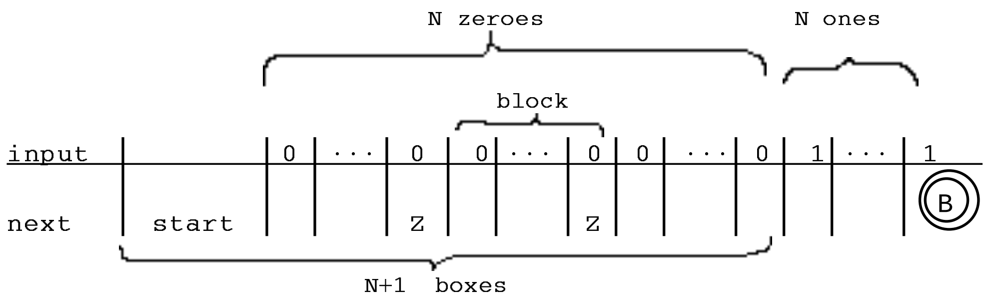


FIG 1

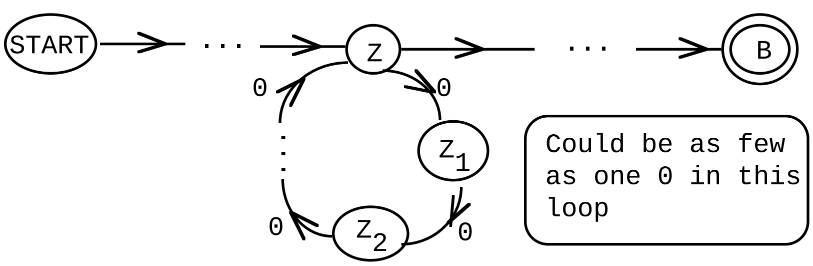


FIG 2

The block of 0's in the loop might contain as few as one 0 if the repeated states are in adjacent boxes in Fig 1. Or it might contain as many as all N of the 0's.

Make a new string by deleting the block of 0's from the old string. The new string still reaches B (same track but without the loop) so it is still accepted.

But the new string has fewer 0's than 1's so it should *not* be accepted. Contradiction!

So A couldn't have been regular after all. QED

example 1 repeated

Instead of *deleting* the block in Fig 1 you could get a new string by *duplicating* the block (duplicating the block once so that it *appears* twice is called pumping the block twice) to get the new string

$$\underbrace{0 \ \ldots \ 0 \text{ block block } 0 \ \ldots \ 0}_{\text{more than N zeroes}} \quad \underbrace{1 \ \ldots \ 1}_{\text{N ones}}$$

The new string still reaches B (on a track that goes around the loop an extra time) so it's accepted. But the new string isn't of the form $0^n 1^n$ since it has more 0's than 1's. So it shouldn't be accepted. Contradiction! So the set is not regular.

example 2 (trickier)

Let A be the set of strings of the form 0^p where p is a prime, i.e.,

$$A = \{0^2, 0^3, 0^5, 0^7, 0^{11}, \ldots\}.$$

I'll show that A is not regular.
Suppose A is regular. (I want to get a contradiction.)
Then it has a FSM recognizer.
Let N be the number of states in the FSM.

Let q be a prime ≥ N. Look at the string 0^q. It's in A so it's recognized by the FSM. Fig 3 shows a track leading to an accepting state.

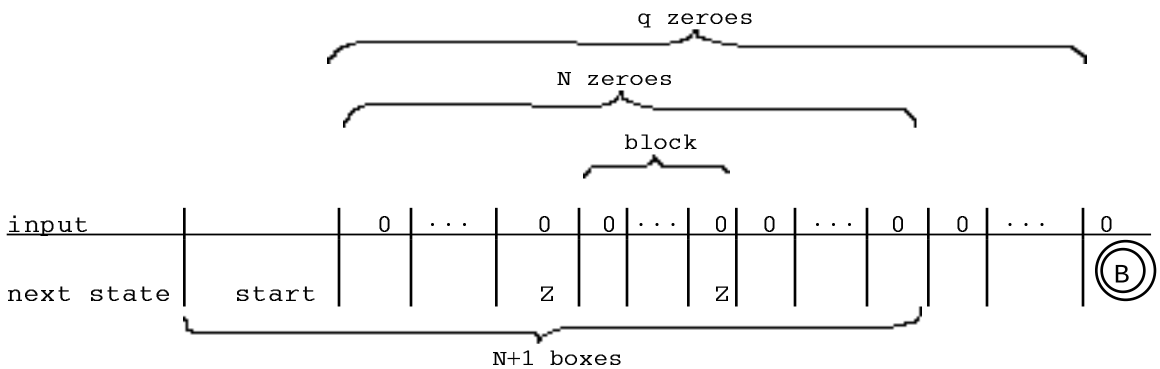


FIG 3

Look at the first N zeroes. There are N+1 corresponding state-boxes in Fig 3, one for each 0 and one for the starting state. More boxes than states. So within the first N zeroes there must be a repeated state, say Z (Fig 3), i.e., the track contains a loop.
Look at the block of 0's in Fig 3 between repeated states. Let the number of 0's in the block be called k. All you know about k is that it's between 1 and N.
Make a new string by duplicating the block q times. The new string still reaches the accepting state B (same track but go around the loop q extra times) so it's accepted. But the new word contains qk *extra* 0's for a total of q + qk zeroes. But

$$q + qk = q(1 + k) \text{ where } 1 + k \geq 2$$

so q + qk factors into two integers neither of which is 1. So q + qk is *not* prime and the new word *shouldn't* be recognized.
Contradiction!.
So the set is not regular.

Note that in this example, it is not enough to duplicate the block once. It had to be done q times to get the contradiction. See problem #3.

example 3

Let A be the set of palindromes (strings which read the same backwards as forwards, e.g., 000, 11011, Madam I'm Adam, Able was I ere I saw Elba).
I'll show that A is not regular.
Suppose it is regular.
Then it has a FSM recognizer, say with N states.

first try (unsuccessful) Look at the palindrome 0^N. Fig 4 shows a track leading to an accepting state. There are N zeroes and N+1 next-state boxes. More boxes than states. So there must be a repeated state, say Z, i.e., the track contains a loop.

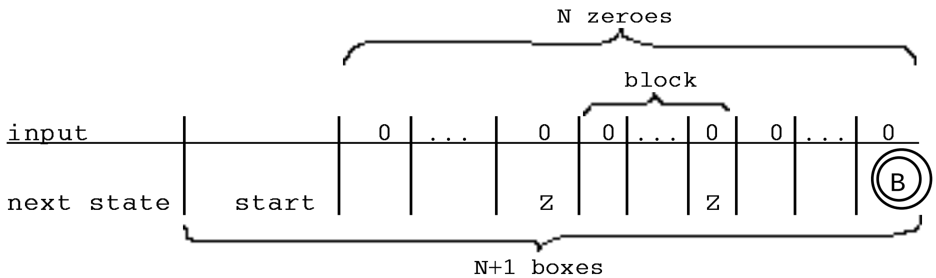


FIG 4

Look at the block of 0's corresponding to the loop. Make a new string by deleting the block or duplicating the block any number of times. The new word still reaches the accepting state B (same track but either skip the loop or go around it more than once) so it's accepted. But this *isn't* a contradiction since the new word is still a palindrome and *should* be accepted. So we have no conclusion (we do *not* have the conclusion that the set is regular).

second try (still unsuccessful) Look at the string $0^{N/2}$ 1 $0^{N/2}$ where N is even (if this works I'll have to come back and consider what to do if N is odd).
Since it is a palindrome, there is a track in the FSM leading to an accepting state (Fig 5). There are N+1 symbols and N+2 next-state boxes. More boxes than states. So there must be a repeated state, say Z.
So the track contains a loop.

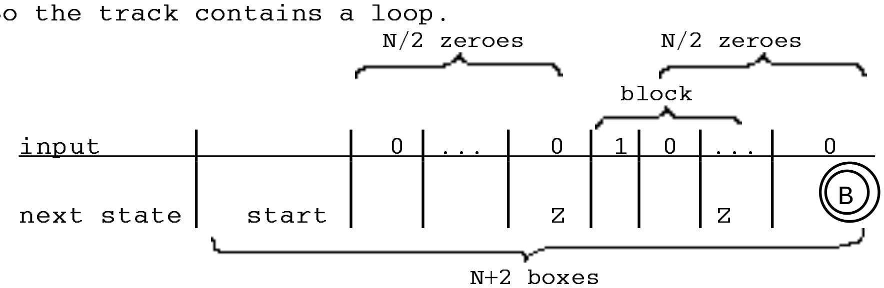


FIG 5

Look at the block corresponding to the loop. It might contain just 0's. It might contain just the 1. It might contains 0's and the 1. Make a new string by deleting the block (or duplicating the block). The new string still reaches B (on a track that skips the loop or goes around it twice) so it's accepted. But *this is not necessarily a contradiction.* If the block consists just of the single 1 then the new string is still a palindrome and deserves to be accepted.

No conclusion about whether the set of palindromes is regular or not.

third try (successful) Look at the string $0^N 1\, 0^N$.
Since it is a palindrome, there is a track in the FSM leading to an accepting state (Fig 6). Within the first N zeroes there are N+1 next-state boxes. More boxes than states so there must be a repeated state, say Z. So the track contains a loop.

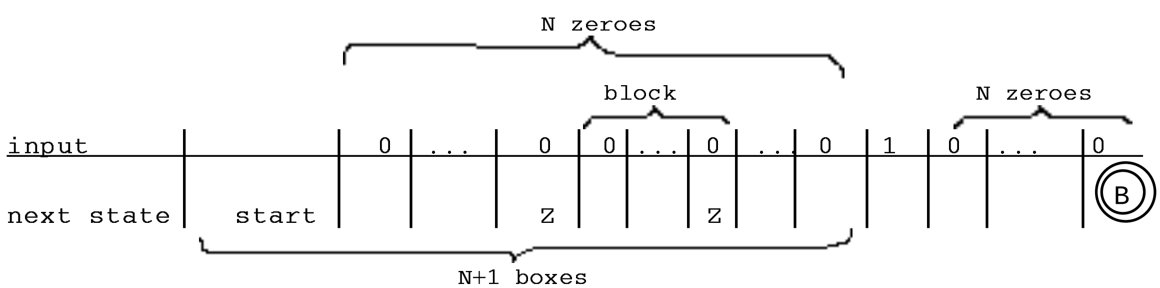


FIG 6

Look at the block of 0's in Fig 6 corresponding to the loop. Make a new string by deleting the block (or duplicating the block). The new string still reaches B (on a track that skips the loop or goes around it twice) so it's accepted. But it isn't a palindrome anymore so it shouldn't be accepted. Contradiction!

So the set of palindromes isn't regular.

outline of the proof you must give to show that a set r is *not* regular

Follow this script as closely as you can.

In particular, you must include first three lines.

Suppose r is regular
Then it has a FSM recognizer.
Let N be the number of states in the FSM.

footnote You need the "Let" because this is a definition of N.

Look at the particular string ··· in r
[Choose a suitable particular string in r to play with; its length must be at least N to be of any use.]

The string has a track leading to an accepting state
There must be a repeated state along the track, i.e., a loop [EXPLAIN WHY].
Make a new string by deleting or duplicating [you decide which] the block of symbols between the repeated states.
The new string still has a track leading to an accepting state but [if you set things up right] the new string isn't in r anymore (doesn't have the "pattern" any more) so it shouldn't be accepted.
Contradiction!
So the set r can't be regular.

warning

1. Not every string in r may help get the contradiction.
In example 3, 0^N didn't help but $0^N 1\, 0^N$ did.

2. You might have to choose a specific amount of duplication to get the contradiction. In example 2 it took q duplications where q was a prime ≥ N.

3. You must work with a *specific* (not vague, not general, not abstract) string in the set and it must have length ≥ N to be useful.

example 4

Let A be the set of binary strings where the number of 0's is a multiple of 3. For example, A contains 111, 10100, 0^9, 010^{11},

Is A regular?

first try (unsuccessful) I'll try to show that A is not regular.
Suppose A is regular.
Then it has a FSM recognizer.
Let N be the number of states.
Look at the string 0^{3N}.
It has a track leading to an accepting state.

In the table in Fig 6 there are 3N zeroes and 3N+1 next-state boxes. More boxes than states so there must be a repeated state, say Z. So the track contains a loop.

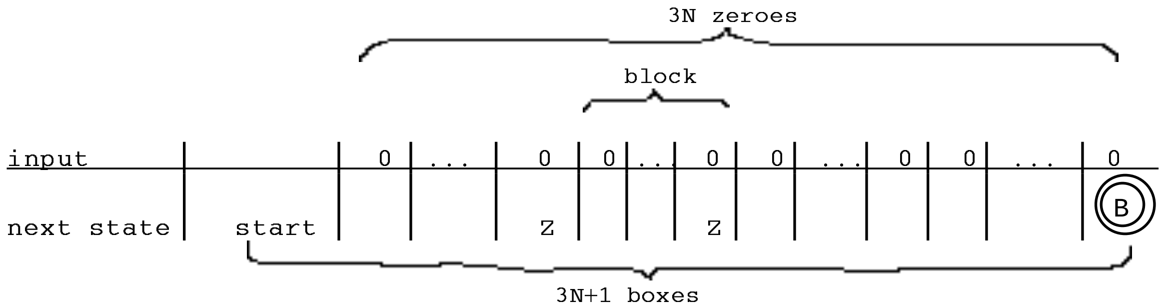


FIG 6

Make a new string by deleting or duplicating the block of 0's in the loop. The new string reaches B (on the same track but without the loop or with extra trips around the loop) so it is still recognized.
But is this a contradiction?
No if the number of 0's in the block happens to be a multiple of 3 in which case the number of 0's in the new string is still a multiple of 3 and the new string deserves to be recognized.
Yes if the number of 0's in the block happens to be 2 or 17 or 5 etc in which case the number of 0's in the new string is no longer a multiple of 3.

But only a *guaranteed* contradiction is useful and we don't have that.
So we have no conclusion about whether A is regular or not.

warning 1 It is of no value to say "I would get a contradiction *if* ... "
You must be able to say "I definitely got a contradiction".

warning 2 When you don't get the contradiction you are looking for it does not mean that the set is regular. It means that you have to start all over again.

second try (WRONG)
Continue as in the first try but delete only *one of the 0's in the block*.
The number of 0's in the new string no longer is a multiple of 3. Contradiction.
So the set is not regular. WRONG WRONG
Here's why the argument is wrong.
To get the contradiction you must know that the number of 0's in the new string is not a multiple of 3 (so far so good, you have that). *And* you must know that the new string reaches an accepting state. You don't know that here. Since you have not deleted *all* the zeros in the block you do *not* have a track that just skips the loop and still reaches B. (You actually have a track that stops one state short of B. That state might be acc also but you can't count on it.)

warning 3 You have to delete or duplicate the entire block corresponding to a loop to have a chance.

third try (successful)
A is regular since it has the regular expression 000(000)*.

fourth try (also successful)
A is regular since it is recognized by the FSM in Fig 7.

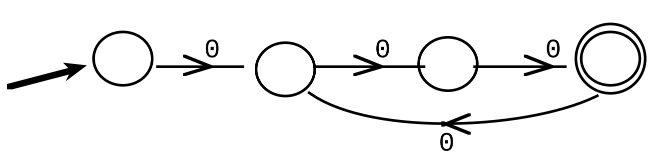


FIG 7

PROBLEMS FOR SECTION 5.7

1. Look at the set of strings of the form $0^n 1^m$ where m > n, i.e., strings of 0's followed by a larger number of 1's. Show that it isn't regular.

2. Look at the set of binary strings where the second half of the string is a repeat of the first half (e.g., 110 110, 10011 10011, 00 00). (The strings must have an even number of symbols to begin with or they don't even have halves.)

Here's the beginning of a proof to show that the set is not regular.

Suppose it is regular.

Then it has a FSM recognizer.

Let N be the number of states.

Continue the argument using the following strings.

(a) $10^N 10^N$

(b) $\underbrace{10 \ldots 0}_{N/2 \text{ symbols}} \; \underbrace{10 \ldots 0}_{N/2 \text{ symbols}}$ (this assumes N is even)

3. Look at example 2 (set of strings with a prime number of 0's). I got a new string by duplicating the block in Fig 3 q times to eventually get a contradiction. Why so complicated? Why can't you just delete the block or duplicate it once? What was so special about duplicating it q times.

4. Let A be the set of binary strings beginning with 0 where every occurrence of 0 is followed immediately by a block of at least three 1's.

Decide if the set is regular or non-regular and prove that you're right.

5. Let A be the set of binary strings not containing any occurrences of 010

Decide if the set is regular or not regular and prove that you're right.

6. Let A be the set of strings of 0's where the number of 0's is a perfect square, i.e., $A = \{0, 0^4, 0^9, 0^{16}, \ldots\}$

Decide if the set is regular or non-regular and prove that you're right.

7. Let r be the set of binary strings with a 1 in the "middle"
Right in the middle if the string has an odd number of symbols and in one of the two middle spots if the string has an even number of symbols.
For example, r contains

001 $\boxed{1}$ 111

0110 $\boxed{1}$0 1000

111 $\boxed{1}$ 111 etc

Let's begin a proof to show that r is not regular.
Suppose it is regular.
Then it has a FSM recognizer.
Let N be the number of states.

(a) What's wrong with the following way to continue the argument.

Look at a string of the form

$$\underbrace{\ldots}_{\text{N symbols}}\ 1\ \underbrace{\ldots}_{\text{N symbols}}$$

The string has a track to an accepting state.
There is a loop within the first N symbols.
Duplicate the block of symbols corresponding to the loop, say ten times for good measure.
The new word is still accepted but the 1 is now pushed way out of the middle so it shouldn't be accepted. Contradiction.
So the set is not regular.

(b) Find a correct way to finish the argument.

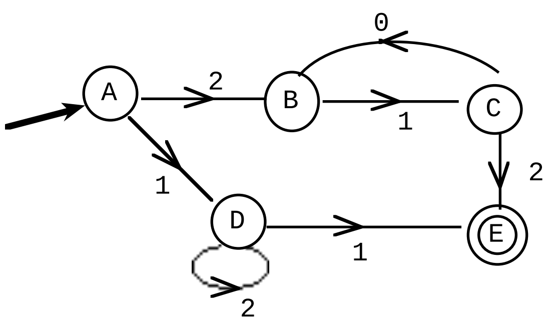
8. Show that the following set of strings is not regular (the alphabet is 0–9).

1
12
123
1234
12345
123456
1234567
12345678
123456789 and now it gets more interesting
123456789 10
123456789 10 11
123456789 10 11 12
123456789 10 11 12 13
etc.

REVIEW PROBLEMS FOR CHAPTER 5

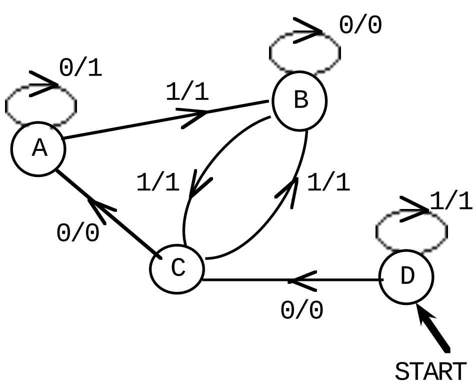
1. The alphabet is a, b, c. Find a FSM recognizer (nondeterministic is fine, but without λ moves) for the strings containing at least one occurrence of bb or at least one occurrence of cc.

2. Solve state equations to find the set of strings recognized by the following FSM. The alphabet here is 0, 1, 2.

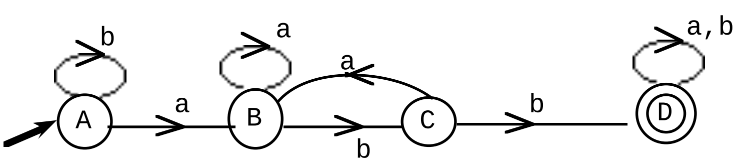


3. Let r be the set of strings with at most two 1's (the alphabet is 0,1). See if you can find three different ways to show that r is regular.

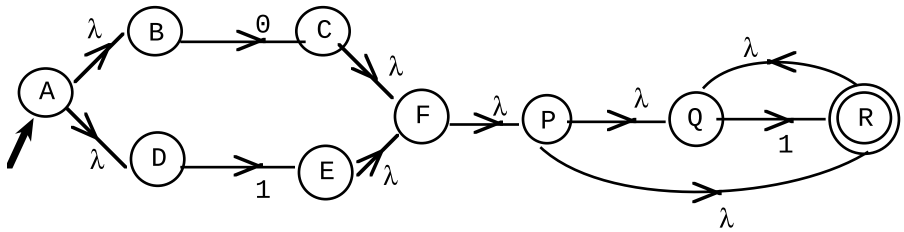
4. Here is a FSM with outputs. Using the standard method to find an equivalent FSM (i.e., one which recognizes the same strings) with accepting states instead of outputs.



5. Solving state equations is a foolproof way to get a regular expression for the set recognized by a FSM and is the guarantee that the set recognized by a FSM is regular. But sometimes you can get a regular expression for the set recognized by a FSM by inspection. Try it for this FSM.

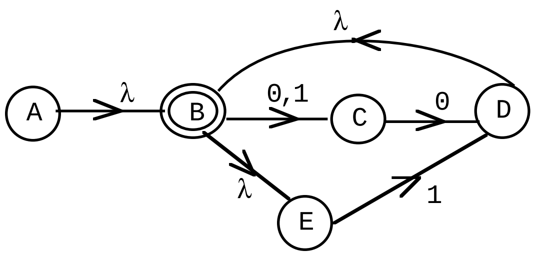


6. Here's a NFSM



(a) By inspection, describe the set of strings recognized.
(b) Find an equivalent DFSM using the standard method.
(c) Find a simpler DFSM by inspection.

7. Keep track of all next states to see if 10 is accepted by this FSM



8. Find a FSM recognizers for $((00+1)^*)^*$ using the Kleene construction method and then find simpler one by inspection.

9. Find a FSM recognizer and a regular expression for the set of strings containing no consecutive 1's.

10. Look at the set of strings of the form $0^3\ 1^m$ where $m \geq 4$, i.e., strings starting with three 0's followed by at least four 1's.

Here's the beginning of a proof trying to show that the set isn't regular.
Suppose the set is regular.
Then it has a FSM recognizer.
Let N be the number of states.
Look at the string $000\ 111\ 1^N$.

Can you finish the proof.

11. Simplify $00^*(0+1)^*1$ if possible

CHAPTER 6 INDUCTION

SECTION 6.1 PROOF BY (WEAK) INDUCTION

principle of mathematical induction

Start with a conjecture about an arbitrary positive integer n like one of these:

The sum of the first n odd integers is n^2.
If a planar connected graph has n edges then V − E + F = 2.

Mathematical induction is a method for proving such conjectures true for all n. (Of course if the conjecture *isn't* true then the attempt to prove it by induction will fail.) The catch is that induction doesn't tell you how to make the conjecture in the first place.

Given a statement involving an arbitrary integer n, the principle of induction says that if you can carry out the following two steps then the statement holds for all $n \ge 1$.
(I) Prove that the statement holds for n = 1
(II) (the inductive step) Prove that if the statement holds for one particular fixed (but arbitrary) value of n, say n=k (this is called the *induction hypothesis*), then it also holds for the value n=k+1.

If you want to prove that a statement is true for say $n \ge 6$ (instead of $n \ge 1$) then, at step I, prove that it holds for n = 6. And continue with the same step II

The induction principle is often viewed as the falling domino principle. If you can topple the first domino (Part I) and if every falling domino knocks over the next one (Part II) then all the dominos fall. Neither I nor II alone is sufficient. Part I without II guarantees only that the first domino falls. Part II without Part I is like being all dressed up with no place to go (no trigger).

example 1

I'll prove that the sum of the first n odd integers is n^2, i.e., that

(1) $$1 + 3 + 5 + \ldots + (2n-1) = n^2 \quad \text{for } n = 1, 2, 3, 4, \ldots$$

Part I Show that it's true for n = 1
Of course it is.
If n = 1 then the LHS of (1) is 1 and the RHS of (1) is 1^2 which is also 1. QED

Part II Assume that it's true for n=k; i.e., assume that for a fixed k,
$$1 + 3 + 5 + \ldots + 2k-1 = k^2 \quad \text{(the induction hypothesis)}$$

Show that it's true for n=k+1; i.e., show that

(2) $$1 + 3 + 5 + \ldots + 2k-1 + 2k+1 = (k+1)^2$$

Here's the proof:

$$1 + 3 + 5 + \ldots + 2k+1 = \underbrace{1 + 3 + 5 + \ldots + 2k-1}_{\substack{\text{this is } k^2 \\ \text{by the induction hypothesis}}} + 2k+1$$

warning The reason here is *not* "by induction" and it is *not* "by part I".
The reason is "by the induction hypothesis", i.e., by the *assumption* in part II that the statement is true for n=k.
Don't forget to state the induction hypothesis at the beginning of part II.

$$= k^2 + 2k+1$$
$$= (k+1)^2$$

This proves part II
And now that I and II are proved, the statement in (1) is true for all n, by induction.

warning In part II, you are not assuming that n = k. You are assuming that *the statement* (which in this case is "the sum of the first n odd integers is n^2) *is true for* $n = k$.

warning about style

In example 1, to prove (2) in part II, it is neither good style nor good logic to write like this:

$$1 + 3 + 5 + \dots + 2n+1 = (n+1)^2$$

$$\underbrace{1 + 3 + 5 + \dots + 2n-1}_{n^2 \text{ by induction hyp}} + 2n+1 = (n+1)^2$$

$$n^2 + 2n + 1 = (n+1)^2$$

$$(n+1)^2 = (n+1)^2$$

TRUE

Don't write like this. You will lose points on an exam if you do.

First of all, it's silly to keep repeating the $(n+1)^2$ on the right side when it would be more efficient to write like this.

$$\underbrace{1 + 3 + 5 + \dots + 2n-1}_{n^2 \text{ by induction hyp}} + 2n+1$$

$$= n^2 + 2n+1$$

$$= (n+1)^2$$

Write like this instead

And secondly, any "proof" in mathematics that *begins* with what you want to prove and *ends* with TRUE is at best badly written and at worst incorrect and *drives me crazy*. If a statement *leads* to something true that does mean that the statement is true.

With this "method" I can prove that 3 = 4:

3 = 4 (start with what you want to prove)
4 = 3 (just reverse the preceding line)
7 = 7 (add the two preceding lines)
TRUE

So conclude that 3 = 4 (??????)

What you *should* do to prove an identity is begin with the lefthand side and work on it until you get the righthand side (as in the second box above), or visa versa. Or work on each side separately until they turn into the same thing. But don't *start* with what you are trying to prove and end up with something like 7 = 7, 1 = 1, $(n+1)^2 = (n+1)^2$ etc. because that kind of mathematical writing is not good style.

example 2

The generalized De Morgan law says that for n = 2, 3, ...,

(3) $$\overline{A_1 \cup A_2 \cup \ldots \cup A_n} = \overline{A_1} \cap \overline{A_2} \cap \ldots \cap \overline{A_n}$$

Here's a proof by induction. (There are also non-inductive proofs.)

Part I Show that (3) holds for n = 2, i.e., show that $\overline{A_1 \cup A_2} = \overline{A_1} \cap \overline{A_2}$.

I think it's obviously true by looking at a Venn diagram.

Both $\overline{A_1 \cup A_2}$ and $\overline{A_1} \cap \overline{A_2}$ turn out to be the shaded area in Fig 1

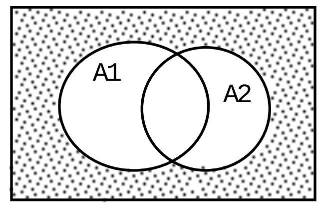


FIG 1

Part II Suppose (3) holds for k sets, i.e., suppose that

$$\overline{A_1 \cup A_2 \cup \ldots \cup A_k} = \overline{A_1} \cap \overline{A_2} \cap \ldots \cap \overline{A_k} \quad \text{(the induction hypothesis)}$$

Now show that (3) holds for k + 1 sets.

$$\overline{A_1 \cup A_2 \cup \ldots \cup A_k \cup A_{k+1}}$$

$$= \overline{(A_1 \cup A_2 \cup \ldots \cup A_k) \cup A_{k+1}} \quad \text{by grouping}$$

$$= \overline{A_1 \cup A_2 \cup \ldots \cup A_k} \cap \overline{A_{k+1}} \quad \text{by part I (the k=2 part)}$$

$$= \left(\overline{A_1} \cap \overline{A_2} \cap \ldots \cap \overline{A_k} \right) \cap \overline{A_{k+1}} \quad \text{by the induction hypothesis}$$

$$= \overline{A_1} \cap \overline{A_2} \cap \ldots \cap \overline{A_k} \cap \overline{A_{k+1}} \quad \text{by ungrouping}$$

So Part II is proved.

example 3

Here's the proof by induction that $5^n - 4n - 1$ is divisible by 16 for n = 1,2,3,...

Part I If n = 1 then $5^n - 4n - 1$ is 0 and 0 *is* divisible by 16

Part II Assume it's true for n=k; i.e., assume that $5^k - 4k - 1$ is divisible by 16 (the induction hypothesis).

> **warning**
> The induction hypothesis is *not* "assume true for all k". It's "assume true for the one particular value n=k".

Try to show that it's true for k+1, i.e., show that $5^{k+1} - 4(k+1) - 1$ is divisible by 16.

By clever algebra you can write $5^{k+1} - 4(k+1) - 1$ in terms of $5^k - 4k - 1$ and make use of the induction hypothesis:

$$5^{k+1} - 4(k+1) - 1 = \text{(by algebra)} \quad \underbrace{5(5^k - 4k - 1)}_{\substack{\text{divisible by 16 by the}\\ \text{induction hypothesis}}} \quad + \quad \underbrace{16k}_{\substack{\text{clearly divisible}\\ \text{by 16}}}$$

So $5^{k+1} - 4(k+1) - 1$ is the sum of two things each of which is divisible by 16 so it's divisible by 16 also. End of part II.

warning

1. In a proof by induction, Part II is *not* to "prove it for n=k+1". In fact if you could prove it for k+1 directly then you wouldn't need an induction proof to begin with. Part II says to prove it for n=k+1 *assuming that it's true for n=k*, which is very different from just proving it for n=k+1.
   Don't forget to have an induction hypothesis in Part II.
2. The induction hypothesis is not "assume true for all k" (as a matter of fact that's what you're trying to prove overall). The induction hypothesis is to assume it's true for a particular (but arbitrary) k.
3. The induction hypothesis is not "assume n = k".
   The induction hypothesis is "assume it is true for n = k" where you fill in the appropriate "it".

PROBLEMS FOR SECTION 6.1

Do everything by induction although other methods of proof may also work and might even be better.
And in each induction proof, state exactly what it is that you're proving in each part.

1. Show that the sum of the first n integers is $\frac{1}{2}n(n+1)$; i.e., show that

$$1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2} \quad \text{for } n \geq 1.$$

2. Show that the sum of the first n squares is $\frac{n(n+1)(2n+1)}{6}$, i.e., show that

$$1^2 + 2^2 + \ldots + n^2 = \frac{n(n+1)(2n+1)}{6} \quad \text{for } n \geq 1.$$

3. Show that $1^3 + 2^3 + 3^3 + \ldots + n^3 = \left[\frac{n(n+1)}{2}\right]^2$ for $n \geq 1$.

4. Show that $11^n - 4^n$ is divisible by 7 for n = 1,2,3,...

5. Show that $2^{2n} - 1$ is divisible by 3 for all positive integers n.

6. A post office sells only 5¢ and 9¢ stamps.
   Show that any postage of 35¢ or more can be paid using only these stamps.

Suggestion In Part II consider two cases for the induction hypothesis:
   case 1 Postage of n¢ can be paid using only 5¢ stamps.
   case 2 Postage for n¢ requires at least one 9¢ stamp.

7. Show that $2^n > n$ for $n \geq 1$.

8. (a) Use induction to show that $n^2 + 5n + 1$ is odd for n = 1, 2, 3, ...
(b) Since part (a) showed that $n^2 + 5n + 1$ is always odd, you couldn't possibly show by induction (or any other method) that it is *even* for all n. But show that part II of the induction argument still works and it's only part I that fails.

9. Show that $(\cos\theta + i\sin\theta)^n = \cos n\theta + i\sin n\theta$ for all $n \geq 1$ where i is the imaginary number satisfying $i^2 = -1$.

For Reference: $\sin(x + y) = \sin x\cos y + \cos x\sin y$
$\cos(x + y) = \cos x\cos y - \sin x\sin y$

10. Show that $\binom{n}{0} + \binom{n}{1} + \ldots + \binom{n}{n} = 2^n$ for $n = 1,2,3,\ldots$

Use Pascal's identity from Section 1.9.

SECTION 6.2 STRONG INDUCTION

principle of strong induction

Given a statement involving an arbitrary integer n, the principle of *strong* induction says that if you can carry out the following two steps then the statement holds for all $n \ge 1$.
(I) Prove that the statement holds for n = 1.
(II) Prove that if the statement holds for n = 1, 2,..., (i.e., not just for n = k but for all integers from 1 up through k) (this is called the *strong induction hypothesis*), then it also holds for n=k+1.

If you want to prove that a statement is true for say $n \ge 6$ (instead of $n \ge 1$) then, at step I, prove that it holds for n = 6. And at step II show that if it holds for n = 6, ..., k then it also holds for n = k+1.

When would you use strong induction rather than ordinary induction?
If you are assuming "it" is true for n = k and that isn't enough to show that it is true for n = k+1, maybe going back and assuming "it" is true for n = 1, 2, ..., k will help. You are entitled to change your induction hypothesis in midstream.

example 1

I want to show that in a planar connected graph, V − E + F = 2.
I'm going to try it by induction on E.

I'll start out using plain induction.

Part I Show that V − E + F = 2 in any planar connected graph where E = 1 (i.e., one edge)
Figs 1 and 2 show the only two planar connected graphs with E=1.

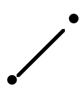


FIG 1

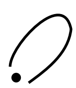


FIG 2

In Fig 1, E = 1, V = 2, F = 1 (the exterior face) so V − E + F = 2 − 1 + 1 = 2
In Fig 2, E = 1, V = 1, F = 2 so V − E + F = 1 − 1 + 2 = 2

Part II Assume that V − E + F = 2 in any planar connected graph with k edges (this is the induction hypothesis).
I want to show that in any planar connected graph with k + 1 edges, V − E + F = 2.

Start with a planar connected graph with k+1 edges.

Delete an edge from this graph so that the new graph has k edges.

case 1 The new graph is still connected.

Imagine that we had deleted say edge BD or edge BC or edge GF in Fig 3.

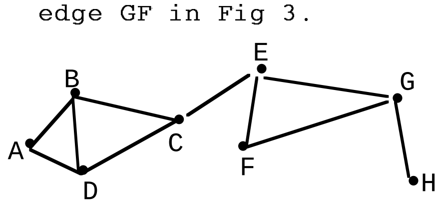


FIG 3

Let E_{new}, V_{new}, F_{new} be the number of edges, vertices, faces in the new graph.

The new graph has k edges, is still connected and of course is still planar. So by the induction hypothesis

(1) $$V_{new} - E_{new} + F_{new} = 2$$

But

(2a) V_{new} = original V

(2b) E_{new} = original E − 1

(2c) F_{new} = original F − 1

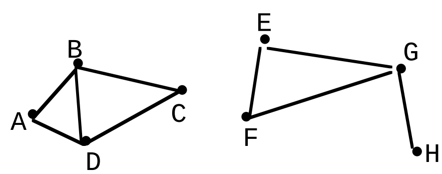
If an edge like BD in Fig 3 had been deleted, two interior faces are thrown together so there is one less face. If an edge like BC is deleted an interior face and the exterior face are thrown together. In each instance, after deleting the edge there is one less face than before.

Substitute (2a,b,c) into (1):

$$V - (E-1) + (F-1) = 2$$
$$V - E + 1 + F - 1 = 2$$
(3) $$V - E + F = 2 \quad \text{QED}$$

case 2 The new graph is not connected.

Look at Fig 3 and imagine deleting say edge CE (Fig 4)



Component 1 Component 2

Fig 4

Then we have two planar connected components.

Let E_1, V_1, F_1 be the number of edges, vertices, faces in Component 1.

Let E_2, V_2, F_2 be the number of edges, vertices, faces in Component 2.

But E_1 is not necessarily k so I can't use the induction hypothesis on Component 1. But I do know that E_1 is between 1 and k. So I will change my induction hypothesis:

Assume that V − E + F = 2 in any planar connected graph where the number of edges is 1, 2, ..., k (strong induction hypothesis).

Now by the strong induction hypothesis,

(4) $$V_1 - E_1 + F_1 = 2$$

And similarly

(5) $$V_2 - E_2 + F_2 = 2$$

Add (4) and (5):

(6) $V_1 + V_2 - (E_1 + E_2) + F_1 + F_2 = 4$

But

(7a) $V_1 + V_2 =$ the original V

(7b) $E1 + E_2 =$ original $E - 1$ (since we deleted an edge)

(7c) $F_1 + F_2 =$ original $F + 1$

> Look at Fig 4 to see that F_1 includes an exterior face and F_2 also includes an exterior face so $F_1 + F_2$ includes an exterior face twice whereas the F from Fig 3 includes the exterior face only once.

Substitute (7a,b,c) into (6):

$V - (E - 1) + F + 1 = 4$

$V - E + 1 + F + 1 = 4$

$V - E + F = 2$ QED

example 1 continued

I can make plain induction work by revising part II a little. (I didn't have to use strong induction.)

Part II Assume that $V - E + F = 2$ in any planar connected graph with k edges (this is the induction hypothesis).
I want to show that in any planar connected graph with k + 1 edges, $V - E + F = 2$.

Start with a planar connected graph with k+1 edges.

case 1 It has a cycle (as in Fig 3 where there are several cycles).
Delete an edge from one of the cycles.
The new graph is still connected: If you had planned to travel from here to there on a path that included the deleted edge you can still travel from here to there by going around the "other" way around the cycle.

> Imagine that in Fig 3 you had planned to go from C to G using path CEG, but edge EG was deleted from the cycle EGFE.. You could still get from C to G on path CEFG

So the new graph is connected, still planar, and has k edges. And we can use the induction hypothesis as in case 1 above (just repeat lines (1)−(3))to conclude that in the original graph (with k+1) edges, $V - E + F = 2$

case 2 It does not have a cycle.
Then the graph is connected, has no cycles, so it is a tree.
We know that in a tree $V = E + 1$ and $F = 1$ so
$V - E + F = E+1 - E + 1 = 2$ QED
(Didn't need any induction hypothesis in this case)

PROBLEMS FOR SECTION 6.2

1. A sequence of numbers is defined as follows:

$F_1 = 1$
$F_2 = 1$
For $n \ge 3$, $F_n = F_{n-1} + F_{n-2}$

Show that

$$(*) \qquad F_n \le \left(\frac{1+\sqrt{5}}{2}\right)^{n-1} \quad \text{for } n = 1,2,3,\ldots$$

using the following version of induction (sort of inbetween weak and strong):

Part I Show that (*) is true for n = 1 *and* n = 2

Part II Show that if (*) is true for n=k−1 *and* n=k then it's true for n=k+1.

Use the algebraic identity $\left(\frac{1+\sqrt{5}}{2}\right)^2 = \frac{1+\sqrt{5}}{2} + 1$

2. The primes are the integers larger than 1 which can't be divided by anything except themselves and 1. For example, 2,5,7,11,13 are the first five primes.

The problem is to show that every integer ≥ 2 is either prime or can be factored into a product of primes (e.g., 12 = 3·2·2)

(a) Try to prove it by ordinary induction but get stuck.

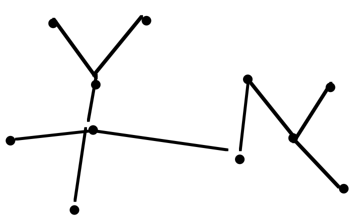
(b) Try it using strong induction.

3. The problem is to use induction on E to show that in a tree, V = E + 1.

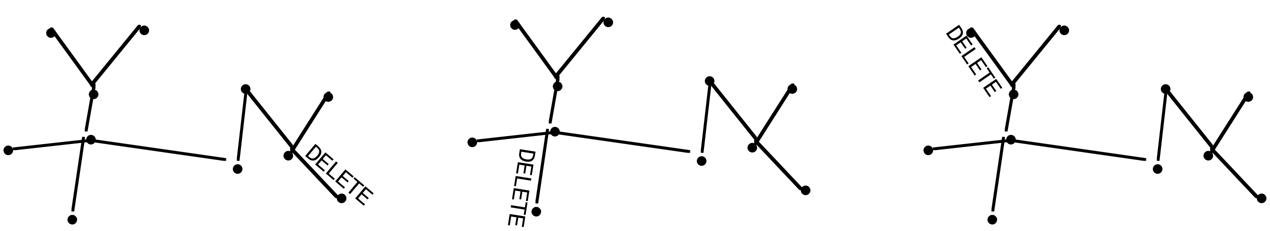
(a) Do Part I.

(b) For Part II you want to show that if V = E + 1 in any tree where E = k (the induction hypothesis) then V = E + 1 in any tree where E = k+1.

To do this, start with a tree with k+1 edges

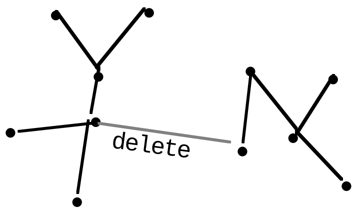


and delete an "outer" edge along with its "outer" vertex too, e.g., like this:



Now keep going.

(c) Suppose that in Part II you start with a tree with k+1 edges and delete an edge that is not an "outer" edge (without removing any vertices) like this:



You are *not* left with one tree with k edges to which the induction hypothesis applies as in part (b). In the diagram above you would be left with two trees each of which has less than k edges.

So ordinary induction would not work.

Do it using strong induction.

The new graph is still connected: If you had planned to travel from here to there on a path that included the deleted edge (e.g., here to A to B to C to there in Fig 5), you can still travel from here to there by going around the "other" way around the cycle (cf. here AGFEDC there)

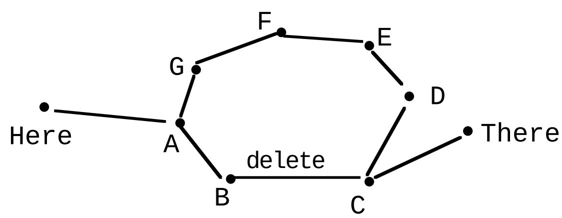


FIG 5

CHAPTER 7 RELATIONS

SECTION 7.1 INTRODUCTION

I'll show you some examples of relations before getting to the formal definition.

example 1

Look at all the courses offered at the university. Let R denote the relation "is a prerequisite for". Then

Physics 1 R Physics 2
Physics 2 R Physics 3
English 1 $\overline{R}$ Physics 1 ($\overline{R}$ means is *not* a prerequisite for) etc.

example 2

Start with the set of all people. Let R denote the relation "is a sibling of". Then

Princess Anne R Prince Charles
Ronald Reagan $\overline{R}$ Queen Elizabeth
Prince William R Prince Harry etc.

example 3

Start with the set of non-negative integers. Let R mean "is 1 larger than". Then

$$4 \; R \; 3, \quad 3 \; \overline{R} \; 4, \quad 4 \; \overline{R} \; 2, \quad 7 \; R \; 6 \quad \text{etc.}$$

definition of a relation

Start with a set A, called the universe.
A relation on A is a collection of ordered pairs (x,y) where x and y are in A.
If (x,y) is in the collection then we write x R y.
If (x,y) is not in the collection then we write x $\overline{R}$ y.

For example, the relation "is 1 more than" defined on the universe of non-negative integers contains the pairs (1,0), (2,1), (3,2) etc.

If A = {a,b,c,d}, some relations on A are

$$R_1 = \{ (a,a), (b,b), (c,c), (d,d) \}$$
$$R_2 = \{ (a,d) \}$$
$$R_3 = \{ (a,b), (b,c), (c,d) \}$$

For instance

$$a \; R_1 \; a, \quad b \; R_1 \; b, \quad a \; R_2 \; d, \quad a \; \overline{R}_2 \; a, \quad a \; R_3 \; b, \quad a \; \overline{R}_3 \; c \quad \text{etc.}$$

In general, if A is a set then the set of *all* ordered pairs (x,y) where x and y are in A is called $A \times A$ so a relation is often referred to as a subset of $A \times A$.

reflexive and antireflexive relations

Let R be a relation on a universe A.
R is called *reflexive* if x R x for all x in the universe
R is called *antireflexive* if x notR x for all x in the universe.

In other words, to decide if a relation is reflexive, look at the universe. For every x in the universe you must have x R x. For antireflexive you must never have x R x.

For example, if the universe is the set of all people in the world then "is the same height as" is reflexive since each person is the same height as herself; "is taller than" is antireflexive" since no person is taller than herself; "thinks highly of" is neither reflexive nor antireflexive since some people think highly of themselves and some don't.

symmetric and antisymmetric relations

A relation R is called *symmetric* if the following holds for all x and y in the universe:

If x R y then y R x.

In other words, for every pair in R, the "reverse" pair is also in R.

I'd like to define an antisymmetric relation as one where pairings never reverse. But with that definition, an antisymmetric relation could never allow x R x since that kind of pairing always reverses. Rather than force an antisymmetric relation to exclude x R x, I'll apply "never reverse" only to pairings x R y where x ≠ y. In other words:

A relation R is called *antisymmetric* if the following holds:

if x ≠ y and x R y then y notR x.

To decide about these properties, look at all the pairs (x,y) in the relation where x ≠ y. If (y,x) is always in there too then the relation is symmetric; if it's never in there then the relation is antisymmetric.

Here's another way to look at it.

Suppose x and y are two different members of the universe.

A symmetric relation includes *either both* (x,y) and (y,x) or includes *neither*.

An antisymmetric relation includes either (x,y) or (y,x) or neither but *not both*.

The relation "is married to" is symmetric since whenever x is married to y then y is also married to x.

The relation "is a subset of" is antisymmetric because for x ≠ y you can't have both x ⊆ y and y ⊆ x.

The relation "is taller than" is antisymmetric because you can't have x taller than y and y taller than x

The relation "admires" is not symmetric (it's possible for x to admire y while y does not admire x) and it's not antisymmetric (it's possible for x to admire y and y to admire x)

transitive relations

A relation R is called *transitive* if the following holds:

if x R y and y R z then x R z.

For example, the relation "is taller than" is transitive: if x is taller than y and y is taller than z then x must be taller than z.

The relation "admires" is not transitive: if x admires y and y admires z then it is not necessarily true that x admires z.

true by default

If you're checking for antisymmetry you have to check that if x R y where x ≠ y then y notR x. If there never is an x R y to begin with then you have no checking to do and the relation is called antisymmetric by default.

And if you're checking for transitivity you have to look at all hookups where x R y and y R z (here, x,y,z don't have to be different) and check that you also have x R z. If there are no hookups at all then there is nothing to check and the relation is called transitive by default.

example 4

Let

A = {5,6,7,8,9} (the universe)

Define a relation R on A as follows: R = { (5,5), (6,6) }

In other words, 5 R 5, 6 R 6 and no other pairs of A are related.

R is not reflexive because you don't have 7 R 7 (or 8 R 8 or 9 R 9).
R is not antireflexive because you do have 5 R 5 (and 6 R 6).
R is symmetric because every pair reverses.

R is antisymmetric by default There's no danger of having both x R y and y R x where x ≠ y because there aren't any of those x R y 's to begin with.
R is transitive because whenever x R y and y R z (which only happens when 5 R 5 and 5 R 5 and when 6 R 6 and 6 R 6) then you also have x R z.

warning

Symmetry does not mean that you always have x R y and y R x. It means that *if* you have x R y then you must also have y R x, i.e., you can't have one pairing without the other. But a symmetric relation can have *neither* x R y *nor* y R x.

example 5

Let the universe be {5,6,7,8} and let R = { (5,6) }, i.e., 5 R 6 and that's all.

R is not reflexive since you don't have 5 R 5 for instance.
R is antireflexive since you never have x R x
R is not symmetric since you have 5 R 6 but not 6 R 5
R is antisymmetric because you don't have the reverse of 5 R 6
R is transitive (by default) since you never have x R y and y R z.

the digraph of a relation

A relation on a set may be pictured as a directed graph.
The vertices represent the elements in the set and an edge is drawn from x to y if x R y
If A = {a,b,c,d,e } and a R b, b R c, c R d then Fig 1 shows the corresponding digraph.
Fig 2 shows the relation "is smaller than" on the set {Dog, Horse, Elephant}

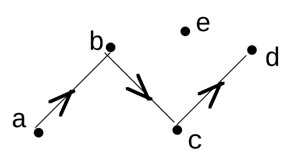


FIG 1

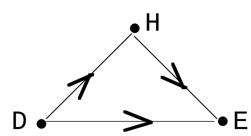


FIG 2

PROBLEMS FOR SECTION 7.1

1. Decide if the relation is reflexive, antireflexive, symmetric, antisymm, transitive.
For the first two, the universe is the set of all people.
For the next three, the universe is the set of integers.
For the last three, the universe is the set of all lines in the plane.

(a) x R y if x is a descendent of y
(b) x R y if x has the same color eyes as y
(c) x R y if xy is even
(d) x R y if x + y < 10
(e) x R y if x < y
(f) x R y if line x has no points in common with line y
(i.e. x is parallel to but not coincident with y)
(g) x R y if x is parallel to or coincident with y
(h) x R y if x is perpendicular to y

2. Define a relation R as follows on the universe {rock, scissors, paper}.
rock R scissors
scissors R paper
paper R rock

Is it reflexive? antireflexive? symmetric? antisymmetric? transitive?

3. Let A = {1,2,3,4}.
Are the following relations on A reflexive? antireflexive? symm? antisymm? trans?
(a) R = { (1,1), (2,1) }
(b) R = { (1,1), (3,3), (4,4) }
(c) R = { (1,2), (2,3), (1,3), (4,2) }

4. Let A = {a,b}.
First decide how many relations can be defined on A and then write them all out.
For each one decide if it is reflexive, antireflexive, symmetric, antisymm, trans

5. Let A = $\{x_1, x_2, x_3, x_4, x_5\}$ be the universe. Consider relations on A.
(a) How many relations are there.
(b) How many reflexive relations are there .
(c) How may relations are there which are reflexive *and* contain (x_2, x_4).
(d) How many symmetric relations are there.
(e) How many relations are symmetric and reflexive.

6. What's special about the digraph of a relation if the relation is
(a) symmetric (b) reflexive (c) transitive

7. If you answer YES then find a specific example.
If you answer NO then explain why not.
(a) Can a relation be both symmetric and antisymmetric.
(b) Can a relation be both reflexive and antireflexive.
(c) Can a relation be neither symmetric nor antisymmetric
(d) Can a relation be neither reflexive nor antireflexive.

SECTION 7.2 EQUIVALENCE RELATIONS

definition and notation

A relation R on a set A is an *equivalence relation* if R is *reflexive, symmetric and transitive*. The generic symbol for an equiv relation is ~

Here are some examples:
is the same age as (for people)
begins with the same letter as (for words)
has the same square as (for integers)
is the same color as (for marbles)

equivalence classes and partitions

Let ~ be an equivalence relation on a set A. If x is in A the *equivalence class* of x is the set of all elements in A equivalent to x. The equivalence class of x contains (at least) x itself since x ~ x by reflexivity.

Look at the equivalence relation "begins with the same letter as". The equivalence class of the word quiet consists of all words beginning with q. The equiv class of the word down consists of all words beginning with d. All in all there are 26 equivalence classes (Fig 1); they are non-overlapping and exhaustive (i.e., every word is in exactly one of them). Fig 1 is called a *partition* of the set of words.

In general a *partition* of a set A is a collection of non-empty disjoint subsets (called cells or blocks of the partition) whose union is A. Fig 2 shows a partition of {a,b,c,d,e,f,g,h,i,j,k,m} with 7 cells.

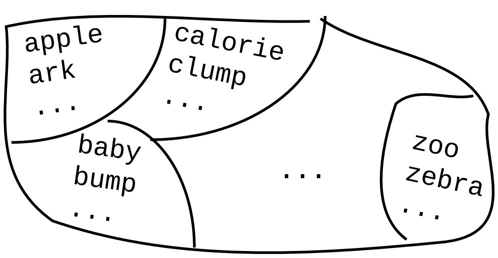


FIG 1

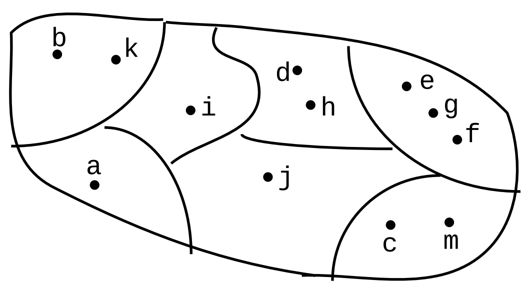


FIG 2

Equivalence relations and partitions are interchangeable ideas as follows:

(1) Every equivalence relation on a set A induces a partition of A
(2) Every partition of a set A induces an equivalence relation on A

In particular if you begin with an equivalence relation then the equivalence classes are a partition of A. For instance, the equivalence relation "begins with the same letter as" produced the partition in Fig 1.

And if you begin with a partition of A then "is in the same cell as" is an equivalence relation. The partition in Fig 2 produces the equivalence relation where b ~ k, d ~ h, e ~ g ~ f, c ~ ℓ (and everything is ~ to itself).

example 1

Look at the equivalence relation "has the same square as" for integers. The equivalence classes are {0}, {−1,1}, {−2,2}, {−3,3} ,..

example 2

Look at the equivalence relation "is the same age as" for children. The equivalence classes are

under a year
1 year olds
2 year olds
⋮
17 year olds

congruence module m (a famous equivalence relation)

Let A be the set of non-negative integers. Let

$$x \; R \; y \text{ if } x-y \text{ is divisible by } 3;$$

i.e.,

$$x \; R \; y \text{ if } x \text{ and } y \text{ have the same remainder when divided by } 3$$

The relation is called *congruence modulo 3*.

For example 14 R 8 (and 8 R 14) because $14 - 8 = 6$ which is divisible by 3 or equivalently because 14 and 8 have the same remainder, namely 2, when divided by 3. Usually we write $14 \equiv 8 \bmod 3$.

Congruence mod 3 is an equivalence relation. There are 3 equivalence classes since every integer is congruent mod 3 to exactly one of 0,1,2. The equivalence classes are

$\{0,3,6,9,12,...\}$
$\{1,4,7,10,13,...\}$
$\{2,5,8,11,14,...\}$.

Similarly $x \equiv y \bmod 6$ if x and y have the same remainder when the y are divided by 6. There are 6 equivalence classes since every integer is congruent mod 6 to exactly one of 0,1,2,3,4,5. The equivalence classes are

$\{0,6,12,18,...\}$
$\{1,7,13,19,...\}$
$\{2,8,14,20,...\}$
$\{3,9,15,21,...\}$
$(4,10,16,22,...\}$
$\{5,11,17,23,...\}$

In general

$$\boxed{x \equiv y \bmod m \text{ if } x \text{ and } y \text{ have the same remainder when they are divided by } m}$$

or equivalently

$$\boxed{x \equiv y \bmod m \text{ if } x - y \text{ is divisible by } m.}$$

There are m equivalence classes since every integer is congruent mod m to exactly one of $0,1,2,...,m-1$.

PROBLEMS FOR SECTION 7.2

1. Is it an equivalence relation? If so describe the equivalence classes. If not, why not?
 (a) same name as
 (b) same major as
 (c) has a class in common with
 (d) has the same two initials as
 (e) lives in the same state as
 (f) lives within 30 miles of
 (g) has the same social security number as
 (h) equals (for integers)
 (i) has the same first four letters as (for words with at least 4 letters)
 (j) has the same y-coordinate as (for points in a 2–dim coord system)
 (k) has the same marital status as
 (ℓ) is married to

2. Let the universe be the set of nonnegative integers.
Is R an equivalence relation?
If so, find the equivalence classes.
(a) n R k if $\sin n\pi = \sin k\pi$
(b) n R k if $\cos n\pi = \cos k\pi$

3. Find all the equivalence relations on {a, b, c} by drawing the corresponding partitions.

4. According to (1), equivalence classes are supposed be be a partition of the universe. So they shouldn't overlap. How can you be so sure they don't.

5. A relation R is called *circular* if it has the following property

If a R b and b R c then c R a

(a) Show that if R is reflexive and circular then R is an equivalence relation.
(b) Show that if R is an equivalence relation then R is circular.

SOLUTIONS Section 1.1

1. The slots are Golf, Tennis, Swimming. Answer is $4\cdot3\cdot6$.

2. There are 4 remaining spots in the number to fill. Each can be filled in 10 ways. Answer is 10^4.

3. Each of the 50 states is a slot which can be filled by one of the two senators from that state. Answer is 2^{50}.

4. (a) Slots are Mon, Tues, ,..., Sun. Each can be filled in 10 ways. Answer is 10^7.
   (b) $10\cdot9\cdot8\cdot7\cdot6\cdot5\cdot4$

5. The slots are the consecutive positions on the sled. Answer is $2\cdot6\cdot5\cdot4\cdot3\cdot2\cdot1$.

6. 2^7 by (2)

7. Pattern must be E_E_E_E_E. Fill in the 4 remaining slots in $4\cdot3\cdot2\cdot1$ ways.

8. (a) The people are the slots. Answer is 11^4.
   (b) $11\cdot10\cdot9\cdot8$.

9. There are 7 slots in the lineup. First 3 are for caps, next 4 for lower case. Answer is $3\cdot2\cdot1\cdot4\cdot3\cdot2\cdot1$.

10. (a) By (2), the answer is 2^{300} (each person is a slot which can be filled in 2 ways, with show or no-show).
   (b) 301 possibilities (0 shows, 1 show,..., 300 shows)

11. Each ball is a slot which can be filled in 10 ways. Answer is 10^{17}.

12. (a) Each traveler is a slot. Answer is $7\cdot6\cdot5\cdot4\cdot3$.
   (b) Each room is a slot (since each room must get a traveler but not vice versa). Answer is $9\cdot8\cdot7\cdot6\cdot5\cdot4\cdot3$.

13. (a) The last three letters are the slots. Answer is 26^3.
(b) There are two slots to fill, each of which can be any letter. Answer is 26^2.
(c) There are 3 slots to fill, each of which must be a non-Z. Answer is 25^3

14. (a) You know that 5 digits are available to be picked but you don't know how many of them are even.
   (b) You know that 5 digits are available but you don't know if they include 0 or not.
   (c) *method 1*
   Find how many end in 0 plus how many end in 2, 4, 6 or 8
   To count how many end in 0:
      Pick last digit. Can be done in 1 way
      Fill the other 5 digits. Can be done in $9\cdot8\cdot7\cdot6\cdot5$ ways
      There are $1\cdot9\cdot8\cdot7\cdot6\cdot5$ of these

   To count how many end in 2,4,6 or 8:
      Pick last digit. Can be done in 4 ways.
      Pick the first digit. Can be done in 8 ways (not 0 and not the one just picked)
      Pick the middle digits. Can be done in $8\cdot7\cdot6\cdot5$ ways.
      There are $4\cdot8\cdot8\cdot7\cdot6\cdot5$ of these.

   So the total number is $1\cdot9\cdot8\cdot7\cdot6\cdot5 + 4\cdot8\cdot8\cdot7\cdot6\cdot5$ [= 68,880]

   method 2
   Total = number that begin with an even digit + number that begin with odd.
   To count how many begin with an even digit:
      Pick first digit. Can be done in 4 ways.
      Pick last digit. Can be done in 4 ways
      Pick the other digits. Can be done in $8\cdot7\cdot6\cdot5$ ways.
      There are $4\cdot4\cdot8\cdot7\cdot6\cdot5$ of these.

To count how many begin with an odd digit:
Pick the first digit. Can be done in 5 ways.
Pick the last digit. Can be done in 5 ways.
Pick the other digits. Can be done in $8\cdot7\cdot6\cdot5$ ways.
There are $5\cdot5\cdot8\cdot7\cdot6\cdot5$ of these.

So the total number is $4\cdot4\cdot8\cdot7\cdot6\cdot5 + 5\cdot5\cdot8\cdot7\cdot6\cdot5$ [= 68,660]

15. To count the total number of names, consider names of length 1, of length 2,..., of length 7 and add.

For names of length 1, there is one spot to fill. It must be done with a letter so there are 26 possibilities.

For names of length 2 there are two spots to fill. The first must be a letter and the second can any of 36 symbols (26 letters, 10 digits). So there are $26\cdot36$ possibilities.

For names of length 3 there are 3 spots to fill. The first must be a letter and the next two spots can be any of 36 symbols. So there are $26\cdot36^2$ names of length 3.

And so on. All in all,

total number of names = $26 + 26\cdot36\cdot + 26\cdot36^2 + 26\cdot36^3 + 26\cdot36^4 + 26\cdot36^5 + 26\cdot36^6$

Why is the answer a *sum*? Because the *total* number of names is the number of names of length 1 *plus* the number of names of length 2 *plus* the number of names of length 3 *plus* etc.

SOLUTIONS Section 1.2

1. (a) Pick a committee of 10 questions from 20. Answer is $\binom{20}{10}$.

   (b) Pick a committee of 6 questions from 16. Answer is $\binom{16}{6}$.

2. (a) Pick a committee of 5 cards from 13 spades $\binom{13}{5}$

   (b) Pick a committee of 5 cards from 16 pictures $\binom{16}{5}$

   (c) Pick 5 cards from the 39 non-hearts $\binom{39}{5}$

   (d) Pick 3 cards from the 50 that are left $\binom{50}{3}$

   (e) Pick 3 cards from the 48 non-aces $\binom{48}{3}$

3. (a) Match up the men and the hats. Can be done in 4! ways.
   (b) someone is dissatisfied = total − everyone is satisfied = 4! − 1
(there is only one outcome satisfying everyone, namely everyone gets his own hat)

4. (a) Fill two slots named King and Queen $10\cdot 7$
   (b) Fill slots named president and vice-pres $17\cdot 16$
   (c) Pick a committee of 2 from 17 $\binom{17}{2}$
   (d) The medals are slots. A person can receive both medals so each slot can be filled in 17 ways. Answer is 17^2.

5. (a) $7\cdot 6 = 42$

   (b) $\dfrac{7!}{2!\ 5!} = \dfrac{7\cdot 6}{1\cdot 2} = 21$

   (c) $\dfrac{8!}{5!\ 3!}\ \dfrac{3!\ 2!}{5!} = \dfrac{8\cdot 7\cdot 6}{5\cdot 4\cdot 3} = \dfrac{28}{5}$

   (d) 12345 (e) 1 (f) 12345 (g) 1

6. $\dfrac{\binom{5}{3}}{\binom{10}{3}} = \dfrac{5!}{3!\ 2!}\ \dfrac{3!\ 7!}{10!} = \dfrac{5\cdot 4\cdot 3}{10\cdot 9\cdot 8} = \dfrac{5}{10}\ \dfrac{4}{9}\ \dfrac{3}{8}$

7. $\dfrac{(n+m-1)!}{(n-1)!\ m!}\ \dfrac{n!\ m!}{(n+m)!} = \dfrac{n}{n+m}$

8. *method 1*

   Use the nannies as the slots. Answer is $10\cdot 9\cdot 8\cdot \ldots \cdot 5 = \dfrac{10!}{4!}$

   method 2

   Pick 6 lucky families and match with the 6 nannies $\binom{10}{6}\cdot 6!$

9. (a) Choose 4 more from the remaining 8. Answer is $\binom{8}{4}$

   (b) *method 1* not-both = total − both = $\binom{10}{6} - \binom{8}{4}$

   method 2 not-both = neither + just A_1 + just A_7 = $\binom{8}{6} + \binom{8}{5} + \binom{8}{5}$

SOLUTIONS Section 1.3

1. Choose 2 from the first 5 in $\binom{5}{2}$ ways. Then pick 8 from the last 15 in $\binom{15}{8}$ ways.

Answer is $\binom{5}{2}\binom{15}{8}$.

2. $\binom{7}{2}\binom{8}{3}$

3. (a) Pick the second spade in 12 ways.

Pick the other 3 cards from the 39 non-spades. Answer is $12\cdot\binom{39}{3}$.

(b) $\binom{13}{3}\binom{13}{2}$

(c) $\binom{26}{4}\cdot 26$

(d) Pick the 2 aces in $\binom{4}{2}$ ways. Pick the 3 other cards from the non-aces in $\binom{48}{3}$ ways. Answer is $\binom{4}{2}\binom{48}{3}$.

(e) Pick the one free card from the 44 non-ace—non-kings. Answer is 44.

4. (a) Pick 3 spots in the license for the digits, (the other 2 spots will automatically get the letters). Then fill.

Answer is $\binom{5}{3}\cdot 10^3\cdot 26^2$.

(b) *method 1*

Like part (a) but fill the spots without replacement.

$\binom{5}{3}\cdot 10\cdot 9\cdot 8\cdot 26\cdot 25$

method 2 Pick 3 digits. Pick 2 letters. Permute them. $\binom{10}{3}\binom{26}{2}\ 5!$

(c) $26^2\cdot 10^3$

5. (a) $\binom{20}{5}$

(b) *method 1*

not both = neither + A and not B + B and not A

$= \binom{18}{5} + \binom{18}{4} + \binom{18}{4}$

method 2 committees with not both = total − both = $\binom{20}{5} - \binom{18}{3}$

(c) neither + both = $\binom{18}{5} + \binom{18}{3}$

6. (a) Pick 5 books for first child, 5 for second etc. Answer is $\binom{20}{5}\binom{15}{5}\binom{10}{5}$

(b) Pick 7 for oldest, 7 for next oldest etc. Answer is $\binom{20}{7}\binom{13}{7}\binom{6}{3}$.

7. (a) Pick 5 from the 15 women and children $\binom{15}{5}$

(b) Pick 4 more from the other 17 people $\binom{17}{4}$

(c) Pick 4 more from the 15 women and children $\binom{15}{4}$

(d) Pick 2 women, then pick 2 more from the non-M_3 men and the children $\binom{7}{2}\binom{10}{2}$

(e) Pick the rest of the committee. $\binom{15}{2}$

(f) Pick a man. Pick 4 non-men $3\cdot\binom{15}{4}$

8. (a) Can pick the 3 men in $\binom{10}{3}$ ways. Can pick the 3 others in $\binom{50}{3}$ ways.

Answer is $\binom{10}{3}\binom{50}{3}$.

(b) *method 1* Take the committee answer from part (a) and multiply by 6! to line it up. Answer is $\binom{10}{3}\binom{50}{3}\cdot 6!$.

method 2 Pick 3 spots in the lineup for the men leaving 3 for the others. Fill the spots. Answer is $\binom{6}{3}\cdot 10\cdot 9\cdot 8\cdot 50\cdot 49\cdot 48$.

9. Pick 25 states from the 50. Then pick one senator from each state $\binom{50}{25}\cdot 2^{25}$

10. *method 1* Pick 3 couples. Then pick one spouse in each couple $\binom{4}{3}\cdot 2^3$

method 2 total − committees with a husband&wife

Total is $\binom{8}{3}$.

For committees with a husband&wife pick one couple (can be done in 4 ways) and then pick another person (can be done in 6 ways). So husband&wife committees = $4\cdot 6$.

Answer is $\binom{8}{3} - 4\cdot 6$.

11. (a) Pick 3 places in the string for the evens, leaving 9 for odds&letters

Then fill the slots. Answer is $\binom{12}{3}\,5^3\,31^9$.

(b) *method 1* $\binom{12}{3}\cdot 5\cdot 4\cdot 3\cdot 31\cdot 30\cdot\ldots\cdot 23 = \binom{12}{3}\,\frac{5!}{2!}\,\frac{31!}{22!}$

method 2 (which works here but wouldn't work for part (a)

Pick 3 evens. Pick 9 from the odds&letters.

Then permute the 12 symbols.

Answer is $\binom{5}{3}\binom{31}{9}\,12!$.

12. The total number of ways she could get all, none or any combination of the 4 items is 2^4 (each item is a slot which can be filled in 2 ways -- she gets it, she doesn't get it) Section 1.1). But subtract the one possibility of getting nothing. Answer is $2^4 - 1$.

13. (a) For 5-letter palindromes, the first 3 spots can be filled in $26\cdot 26\cdot 26$ ways and the other spots are then determined. Answer is 26^3.

Same answer for 6-letter palindromes

(b) For a 5-letter palindrome, the first spot can be filled in 26 ways, the second in 25 ways, third in 24 ways. Last spots are determined. Answer is $26\cdot 25\cdot 24$

Same answer for 6-letter palindromes

14. (a) $\binom{15}{4}$

(b) $15\cdot 14\cdot 13\cdot 12$

(c) $15\cdot 14\cdot 13\cdot 12$ (use the books as slots)

(d) 15^4

(e) I assume that someone who is on the team can't also do the cheering. So pick 5 from 15 to play and then pick 4 from the remaining 10 to cheer.

Answer is $\binom{15}{5}\binom{10}{4}$ or equivalently $\binom{15}{4}\binom{11}{5}$.

(f) I assume that a person *can* be picked to play both. So pick 5 from 15 to play basketball and then pick 4 from 15 to play football. Answer is $\binom{15}{5}\binom{15}{4}$.

(g) Pick 9 from 15; then pick 2 from 6. The remaining 3 are automatically the cheerers. Answer is $\binom{15}{9}\binom{6}{2}$

15. Pick the ace, then pick 4 others, $4\binom{48}{4}$

16. Pick 3 spots for the B's. Then the entire permutation is determined since the B's go in those 3 spots in ascending order and the A's go in the remaining 4 spots in

ascending order. Answer is $\binom{7}{3}$.

Or, equivalently, pick 4 spots for the A's and get answer $\binom{7}{4}$. Same as $\binom{7}{3}$.

17. (a) *method 1* Use the women as slots $9\cdot 8\cdot 7\cdot 6\cdot 5$

method 2 Pick 5 men. Match them with the 4 women $\binom{9}{5}$ 5!

(b) In how many ways can a coed couple be picked. In other words, 5'·9 counts this list:

(1) $W_2\ M_7$

(2) $W_2\ M_2$

(3) $W_3\ M_1$

etc.

18. There are 4! ways to arrange the letters D,E,F and the chunk ABC. Then there are 2 acceptable ways to arrange the chunk (ABC and CBA). Answer is $4!\cdot 2$

19. 2^5 (Each topping slot can be filled with yes or no. This includes the possibility of a bare hamburger; i.e., this is an all, none, any combination type of problem.)

20. (a) Pick one remaining card. Answer is 48.

(b) Pick a face value for the 4-of-a-kind. Then pick a fifth card. Answer is $13\cdot 48$.

21. (a) *method 1* Use the people as slots. Answer is $7\cdot 6\cdot 5\cdot 4$

method 2 Pick 4 seats. Match them with the 4 people. Answer is $\binom{7}{4}$ 4!

(b) Pick a block of 4 consecutive seat.

Can be done in 4 ways (S_1-S_4, S_2-S_5, S_3-S_6, S_4-S_7)

Then match the people with the seats. Answer is $4\cdot 4!$.

22. *method 1* Choose the chairperson. Then choose 4 others. $15\cdot\binom{14}{4}$

method 2 Choose a committee of size 5. Then select one of them to be chair. Answer is $\binom{15}{5}\cdot 5$.

23.(a) Permute 10 boys and a girl sextet. Then permute within the sextet. 11! 6!

(b) Permute the 10 boys. Then each girl is a slot which can be filled with one of the 11 spaces between or at the ends of the boys. Answer is $10!\cdot 11\cdot 10\cdot 9\cdot 8\cdot 7\cdot 6$

method 2 Permute the 10 boys. Pick 6 of the 11 between/ends. Match the girls to those 6 between/ends. Anaswer is 10! $\binom{11}{6}$ 6!

(c) total – answer to (a) is the number of ways of lining them up so that not all six girls are together. But it still counts lineups where 5 of the girls are together or 4 are together etc. It doesn't keep the girls totally apart.

In other words, the opposite of "no two girls together" is *not* "girls all together".

24. (a) Line up a D block, an R block, an S block and 7 others. Then permute within the blocks. Answer is 10! 3! 5! 6!.

(b) The only thing left to do is line up the people within each groups. Answer is 3! 5! 6! 7!.

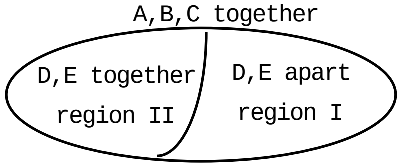
25. (a) Line up the 23 other letters together with the ABC clump. Then permute within the clump. Answer is 24! 3!.

(b) Line up the other 24 letters. Then pick one of the 25 between/ends for the D and another for the E. Answer is $24!\ 25\cdot 24$.

(c) Permute an ABC clump, a DE clump and the 21 other letters. Then permute within each clump. Answer is 23! 3! 2!.

(d) *method 1* Line up the ABC clump and the 21 letters F-Z. Can be done in 22! ways.
Then permute within the clump. Can be done in 3! ways.
Then choose one of the 23 between/ends for the D and another for the E.
Answer is $22!\,3!\,23\cdot 22$.

method 2
Look at the Venn diagram below:
A,B,C together & D,E apart (region I)
= A,B,C together − A,B,C together & D,E together (region II)
= answer to (a) − answer to (c)
= $24!\,3! - 23!\,3!\,2!$



26. (a) Each set is a slot which can be filled with J or M. Answer is 2^5.
(b) Mary can win anywhere from 0 to 5 sets. Answer is 6.

27. Permute the 21 objects consisting of the string mother and the remaining 20 letters. Answer is 21!

28. Pick the face value. Then pick 3 cards from that face. $13\cdot\binom{4}{3}$

29. (a) Pick 2 faces for the 2 pairs. Then pick 2 from each face. $\binom{13}{2}\binom{4}{2}\binom{4}{2}$
(b) Pick a face for the 3-of-a-kind and then a face for the pair. Then pick 3 cards from the first face and 2 from the second face. $13\cdot 12\cdot\binom{4}{3}\binom{4}{2}$

Question Why does part (a) have $\binom{13}{2}$ while part (b) has $13\cdot 12$?
Answer In (a) you need 2 faces, one for each pair, but they can't be used as slots because there isn't a first pair and a second pair. So in (a) you need a committee of 2 faces. Getting a pair of Queens and a pair of Jacks is the same as getting a pair of Jacks and a pair of Queens.
In (b) you need a face specifically for the 3-of-a-kind and a face for the pair. Getting 3 Queens and 2 Jacks is different from getting 2 Queens and 3 Jacks. In (b) you can pick a committee of 2 faces but then you have to multiply by 2 to decide which face will get the 3-of-a-kind and which will get the pair. So you either have to use $\binom{13}{2}\cdot 2$ or $13\cdot 12$.

30. (a) There are 2^{50} subsets (Section 1.1, page 3). Throw away the null subset and the subset consisting of all 50. Answer is $2^{50} - 2$.

(b) comms of size 2 + comms of size 3 + ... + comms of size 6
$$= \binom{50}{2} + \binom{50}{3} + \binom{50}{3} + \binom{50}{4} + \binom{50}{5} + \binom{50}{6}$$

31. (a) A line is determined by (a committee of) two points. There are $\binom{23}{2}$ pairs of points. Since no 3 points are collinear each of the $\binom{23}{2}$ pairs determines a different line. Answer is $\binom{23}{2}$.
(b) Might be as many as $\binom{23}{2}$ if no 3 points are collinear and might be as few as 1 if all the points are collinear. Can't pin it down any further.

32. (a) $7 \cdot 7 \cdot 7$

(b) $7 \cdot 6 \cdot 5$

(c) $\binom{7}{3}$

(d) With replacement, unordered. A committee where someone can serve more than once. For example $A_2A_2A_5$ is one such sample and is the same as $A_2A_5A_2$ since order doesn't count.

33. (a) This method counts the following as different outcomes when they are really the same.

outcome 1

step 1 Pick J_S, J_H, J_C

step 2 Pick A_S

step 3 Pick 3_H

outcome 2

step 1 Pick J_S, J_H, J_C

step 2 Pick 3_H

step 3 Pick A_S

In general this method counts every outcome twice.

Here's the correct version.
Pick a face value and 3 cards from that value.
Pick two more faces from the remaining 12 (a *committee* of 2 faces).
Pick a card in each of those 2 faces.
Answer is $13 \cdot \binom{4}{3}\binom{12}{2} \cdot 4^2$.

(b) This method counts the following as difference outcomes when they are really the same.

outcome 1 step 1 Pick spot 1 to get the sure z.
step 2 Pick q for spot 2, pick z for spot 3.

outcome 2 step 1 Pick spot 3 to get the sure z.
step 2 Pick z for spot 1, q for spot 2.

In general this method counts words with one z once each, counts words with two z's twice each and counts the word zzz three times

Here's a correct version.

contains z = total − no z's = $26^3 - 25^3$

(c) It counts the following outcomes as different when they are really the same.

outcome 1 Pick spot 1 for an A.
Pick spot 2 to get an A.
Pick spot 7 to get an A.
Fill other spots with Z's

outcome 2 Pick spot 2 for an A.
Pick spot 1 for an A.
Pick spot 7 for an A.
Fill other spots with Z's.

Here's a correct method.
Pick a committee of 3 spots for the A's.
Fill the other places from the remaining 25 letters.
Answer is $\binom{7}{3} \cdot 25^4$.

(d) It counts the hand $J_H 3_S$ as different from the hand $3_S J_H$. In fact it counts every outcome exactly twice. It uses "first card" and "second card" as slots but there is no such thing as a first or a second card in a hand.

Here are some correct versions.

method 1

Pick a committee of 2 faces. Pick a card from each face.

Answer is $\binom{13}{2} \cdot 4^2$ [= 1248]

method 2

Take half the wrong answer.

method 3

not containing pair = total − contains pair.

The total number of 2-card hands is $\binom{52}{5}$

To find the number containing a pair:

Pick a face for the pair. Can be done in 13 waqys.

Pick 2 cards in that face. Can be done in $\binom{4}{2}$ ways.

So there are 13 $\binom{4}{2}$ hands with pairs.

Final answer is $\binom{52}{5} - 13 \binom{4}{2}$

(e) It counts the following outcomes as different when they are the same

outcome 1 Step 1 M_1 W_2 Step 2 M_6 W_4 Step 3 M_5 W_3

outcome 2 Step 1 $M_6 W_4$ Step 2 M_1 W_2 Step 3 M_5 W_3

The method uses "first couple", "second couple", "third couple" as slots when the couples are not designated first, second, third.

Here are some correct methods.

method 1 Pick 3 men, pick 3 women, match the men and women. Answer is $\binom{10}{3}\binom{8}{3}$ 3!.

method 2 Pick 3 men. Then use each man as a slot and fill with women. Answer is $\binom{10}{3} 8 \cdot 7 \cdot 6$

method 3 The wrong answer counted each outcome precisely 3! times so divide by 3! to get the right answer $\dfrac{10 \cdot 8 \cdot 9 \cdot 7 \cdot 8 \cdot 6}{3\,!}$.

SOLUTIONS Section 1.4

1. $\frac{15!}{3!\ 7!\ 5!}$

2. (a) $6!$

(b) There are 2 O's and 2 L's. Answer is $\frac{8!}{2!\ 2!}$.

3.(a) Permute these seven things: 2 L's, 2 C's, G, S and a vowel clump.
Then permute the clump which contains 3 O's, 2 I's and an A.

$\frac{7!}{2!\ 2!}\ \frac{6!}{3!\ 2!}$

(b) Pick 6 spots for the vowels (then the vowels automatically fill these spots in the order AIIOOO). Permute the 6 consonants for the remaining 6 spots.

$\binom{12}{6}\ \frac{6!}{2!\ 2!}$

(c) Like (a) but now there is only one way to arrange the vowel clump (in the order AIIOOO). Answer is $\frac{7!}{2!\ 2!}$.

4. (a) It double counts. For instance it counts the following as different outcomes when they are really the same.

outcome 1

step 1 Pick A
step 2 Pick D
step 3 Line them up ADDA

outcome 2

step 1 Pick D
step 2 Pick A
step 3 Line up them ADDA

Every outcome is counted precisely twice.

(b) Pick the two letters in $\binom{26}{2}$ ways, not in $26 \cdot 25$ ways, because you want a committee of two letters not a *first* letter and a *second* letter.

Answer is $\binom{26}{2}\ \frac{4!}{2!\ 2!}$.

Or you can take the wrong answer and divide by 2.

5. After a plus is put in the middle you have 3 pluses, 4 minuses and 3 crosses to permute. Answer is $\frac{10!}{3!\ 4!\ 3!}$.

6. (a) 3^{10} (each spot in the word can be filled in 3 ways)

(b) $\frac{10!}{3!\ 4!\ 3!}$ (permute 3 A's, 4 B's, 3 C's)

(c) Pick 3 places for the A's. Then fill the other 7 places with B or C. $\binom{10}{3} \cdot 2^7$

7. (a) Permute the 2 P's, A,L,R,T. Then pick 3 of the 7 between/ends for the E's.
Answer is $\frac{6!}{2!}\ \binom{7}{3}$.

(b) Permute the 2 P's, 3 E's, T. Can be done in $\frac{6!}{2!\ 3!}$ ways.

Then put A, L, T in 3 of the 7 between/ends..
One way to do this is to make A,L,T each a slot and fill then in $7 \cdot 6 \cdot 5$ ways
Another way is to pick 3 of the 7 spaces and match them to A,L,T.
Can be done in $\binom{7}{3}\ 3!$.

Answer is $\frac{6!}{2!\ 3!}\ 7 \cdot 6 \cdot 5$ or equivalently $\frac{6!}{2!\ 3!}\ \binom{7}{3}\ 3!$.

(c) Line up these 4 things: an ALT clump, 2 P's and an R. Can be done in $\frac{4!}{2!}$ ways.
Then permute the clump. Can be done in 3! ways.
The 4 things have 5 betwee/ends.
Pick 3 of the 5 for the E's. Can be done in $\binom{5}{3}$ ways.

Answer is $\frac{4!}{2!}\,3!\,\binom{5}{3}$.

SOLUTIONS Section 1.5

1. Toss 1000 identical balls (the tickets) into 4 boxes. $\binom{1000+4-1}{1000} = \binom{1003}{1000}$.

2. This is like tossing 19 identical balls into 6 distinguishable boxes.
Answer is $\binom{19+6-1}{19} = \binom{24}{19}$.

3. (a) I'm counting this list:
3 pennies, 10 nickels, 7 dimes
1 penny, 1 nickel, 1 dime, 17 quarters
20 pennies
20 nickels
20 dimes
19 pennies 1 quarter
etc.

Choose a committee of size 20 from a population of size 4 (penny, nickel, dime, quarter), repetition allowed. $\binom{20+4-1}{20} = \binom{23}{20}$

(b) Large means "contains at least 20 coins".
The list I counted in part (a) included the possibilities of all pennies, all nickels, all dimes etc. It assumed that each box contained at least 20 coins. If any of the boxes contained fewer than 20 coins the list of possibilities would be shorter (different game).

4. (a) $\binom{12+5-1}{12}$

(b) Each ball is a slot which can be filled in 5 ways. Answer is 5^{12}

(c) Toss the red balls. Toss the white balls. Answer is $\binom{8+5-1}{8}\binom{4+5-1}{4}$

5. The dime, nickel, and quarter can each be given out in 5 ways.
The 25 pennies are indistinguishable balls which can be tossed into the 5 people in $\binom{25+5-1}{25}$ ways. Answer is $5^3 \cdot \binom{29}{25}$.

6. (a) (ordinary committee) $\binom{10}{6}$

(b) (committee on which a ball can appear more than once) $\binom{10+6-1}{6}$

7. (a) Each course is a slot which can be filled in 5 ways. Answer is 5^6.
(b) Choose a committee of size 6 from the population A, B, C, D, E with repetition allowed. Answer is $\binom{6+5-1}{6}$.

8. (a) Choose a committee of size 6 from a population of size 5 with repetition allowed. $\binom{6+5-1}{6}$

Note that the numbers 7, 8, 9, 10, 12 (how many of each model the dealer has) are almost irrelevant. All that matters is that each is ≥ 6 so that you can choose 6 of any one model if you like (i.e., the dealer is not going to run out of any model).

(b) Choose the 2 models in $\binom{5}{2}$ ways). Say they are M_1 and M_4.
Then the committee of size 6 must include an M_1 and an M_4 (at least one of each) and none of the other models so you have to choose 4 more from a population of size 2. This can be done in $\binom{4+2-1}{4}$ ways.

Answer is $\binom{5}{2}\binom{5}{4}$.

(c) You have to order 2 of 1 model and 1 each of the other 4 models. The only choice you have is which model to double up on. Can do this in 5 ways. Answer is 5.

9. (a) *Step 1* Toss the identical bats into the 20 children.

Can be done in $\binom{6+20-1}{6}$ ways.

Step 2 Toss the balls into the children. Can be done in $\binom{7+20-1}{7}$ ways.

Answer is $\binom{6+20-1}{6}\binom{7+20-1}{7}$.

(b) Pick a committee (a plain committee, with no repetition) of 6 children to get the bats. Can be done in $\binom{20}{6}$ ways.

Pick a committee of 7 children to get the balls. Can be done in $\binom{20}{7}$ ways.

Answer is $\binom{20}{6}\binom{20}{7}$.

(c) *method 1* Pick a committee of 6 children to get the bats. Then pick 7 of the remaining children to get the balls. Answer is $\binom{20}{6}\binom{14}{7}$

method 2 Pick 13 children to get stuff. Can be done in $\binom{20}{13}$ ways.

Then pick 6 of those 13 to get bats. Can be done in $\binom{13}{6}$ ways.

Answer is $\binom{20}{13}\binom{13}{6}$.

10. The symbols can be arranged in 12! ways.
There are 11 spaces between the symbols. After putting 3 blanks in each of these 11 spaces you have 17 blanks (indistinguishable balls) to toss into the 11 spaces (distinguishable boxes). Can be done in $\binom{17+11-1}{17}$ ways. Answer is $12!\binom{27}{17}$.

11. (a) A session is a committee of size 10 chosen from a population of size 4, repetition allowed of course. There are $\binom{10+4-1}{10}$ of them.

(b) After playing each piece twice, the remainder of the session is a committee of size 2 chosen from the 4 pieces, repetition allowed. Answer is $\binom{2+4-1}{2} = \binom{5}{2} = 10$

(c) Here are the 10 possibilities. Each entry in the table is the number of times the piece is played in that session.

		session									
		1	2	3	4	5	6	7	8	9	10
piece	P_1	4	2	2	2	3	3	3	2	2	2
	P_2	2	4	2	2	3	2	2	3	3	2
	P_3	2	2	4	2	2	3	2	2	3	3
	P_4	2	2	2	4	2	2	3	3	2	3

(d) Include 4 P_3's. Then choose 6 more from the non-P_3's. Answer is $\binom{6+3-1}{6}$.

(e) total − 4 or more P_2's

To do "4 or more P_2's" put 4 P_2's on the committee and pick 6 more from $P_1 - P_{10}$ (allowing more P_2's)

Answer is $\binom{10+4-1}{10} - \binom{6+4-1}{6}$

12. (a) Give each box one ball and toss the 7 others into the 5 boxes $\binom{7+5-1}{7}$

(b) 1 ball in B_2 + 3 balls in B_2 + ... + 9 balls in B_2 + 11 balls in B_2

$= \binom{11+4-1}{11} + \binom{9+4-1}{9} + \binom{7+4-1}{7} + \binom{5+4-1}{5} + \binom{3+4-1}{3} + \binom{1+4-1}{1}$

(c) Suppose the 3 boxes chosen at step 1 are B_1, B_2, B_4.

When you toss the 12 balls into these boxes at step 2 you might get all 12 balls into B_1. In that case you would have 4 empties overall instead of 2.

Or you might have 7 in B_2, 5 in B_4 in which case you would have 3 empties overall instead of 2. etc.

The correct step 2 is to put a ball in each of the 3 boxes chosen at step 1 to make sure each one gets something (to make sure you don't get more than 2 empties) and toss the remaining 9 balls into the 3 boxes. Can be done in $\binom{9+3-1}{9}$ ways.

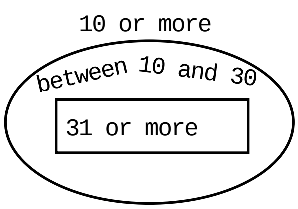
Final answer is $\binom{5}{3}\binom{9+3-1}{9}$

13. This problem is about committees of size 100 chosen from a pop of 5 (A, B, C, D, E), repeated members allowed.

Between 10 and 30 A's on committee = at least 10 − at least 31

For "at least 10", put 10 A's on the committee and choose the 90 others from the 5 (allowing more A's).

Answer is $\binom{90+5-1}{90} - \binom{69+5-1}{69}$.



footnote The long sum answer is

$$10 \text{ A's} + 11 \text{ A's} + \ldots + 30 \text{ A's} = \sum_{n=70}^{90} \binom{n+4-1}{n}$$

14. The ice cream order is a committee of 12 flavors chosen from a population of 31.

(a) No repetition allowed. It's an ordinary committee. $\binom{31}{12}$

(b) Include 6 chocs. Pick 6 more (repetition allowed) from non-choc flavors.

Answer is $\binom{6+30-1}{6}$

(c) First choose 7 flavors out of 31. Can be done in $\binom{31}{7}$ ways.

Now get one cone in each of the 7 chosen flavors (if you don't do this you may end up with fewer than 7 flavors) and then choose 5 more cones from the 7 flavors. Can be done in $\binom{5+7-1}{5}$ ways.

Answer is $\binom{31}{7}\binom{11}{5}$.

(d) total − 3 or more strawberries

For "3 or more straw", put 3 strawberries on the committee and pick 9 more from the 31 flavors. Answer is $\binom{12+31-1}{12} - \binom{9+31-1}{9}$

15. This is tossing 13 indistinguishable balls into 3 distinguishable boxes named x1, x2, x3.

(a) $\binom{3+13-1}{13}$

(b) Box x_1 gets at least 4 balls and box x_2 gets at least 2 balls.

Put 4 balls into x_1, put 2 balls into x_2 and toss the other 7 balls.

Answer is $\binom{7+3-1}{7}$.

(c) *method 1*

Between 2 and 5 into box x3 = 2 or more − 6 or more

For "2 or more" put 2 into box x3 and toss the other 11 into the 3 boxes, etc.

Answer is $\binom{11+3-1}{11} - \binom{7+3-1}{7}$ [= 42]

method 2

Toss 13 indist ball into 5 boxes.

Between 2 and 5 balls in box x3

= exactly 2 + exactly 3 + exactly 4 + exactly 5

$= \binom{11+2-1}{11} + \binom{10+2-1}{10} + \binom{9+2-1}{9} + \binom{8+2-1}{8}$

(d) Now box x_1 gets at least 4 balls, box x_2 gets at least 2 balls and box x_3 gets between 2 and 5 balls.

Put 4 balls into box x_1.

Put 2 balls into box x_2.

Put 2 balls into box x_3.

The problem boils down to tossing the other 5 balls into the 3 boxes so that x_3 gets at most 3 of them. Use total − opp.

The opp is "4 or more into x_3"

To count the opposite event, put 4 of the 5 balls into x_3 and toss the remaining ball -- officially can be done in $\binom{1+3-1}{1}$ ways; obviously can be done in 3 ways.

Answer is $\binom{5+3-1}{5} - \binom{1+3-1}{1}$.

SOLUTIONS Section 1.6

1. N(2A or 3K) = N(2A) + N(3A) − N(2A and 3K) = $\binom{4}{2}\binom{48}{3} + \binom{4}{3}\binom{48}{2} - \binom{4}{2}\binom{4}{3}$

2. *method 1* $N(A_S) + N(K_S) - N(A_S \& K_S) = \binom{51}{4} + \binom{51}{4} - \binom{50}{3}$

method 2 total − N(no spade ace and no spade king) = $\binom{52}{5} - \binom{50}{5}$

3. N(two 3's or two 6's) = N(two 3's) + N(two 6's) − N(two 3's and two 6's)
To find N(two 3's) pick two places for the 3's and then fill the other places from the non-3's.
Similarly for N(two 6's).
To find N(two 3's and two 6's), pick two places for the 3's, two of the remaining places for the 6's and fill the other 3 spots from non-3's-non-6's.
Answer is $\binom{7}{2} 9^5 + \binom{7}{2} 9^5 - \binom{7}{2}\binom{5}{2} 8^3$.

4. (a) Counting the 2-at-a-time terms is like counting twosomes from a population of size 8. Answer is $\binom{8}{2}$.

(b) This is like forming all possible committees of size 3. Answer is $\binom{8}{3}$.

5. (a) N(2A) + N(2K) − N(2A and 2K) = $\binom{4}{2}\binom{48}{3} + \binom{4}{2}\binom{48}{3} - \binom{4}{2}\binom{4}{2}44$

(b) The event 3A and the event 3K are mutually exclusive. So all you need is
N(3A) + N(3K) = $\binom{4}{3}\binom{48}{2} + \binom{4}{3}\binom{48}{2}$

6. N(no S or no H) = N(no S) + N(no H) − N(no S and no H) = $\binom{39}{5} + \binom{39}{5} - \binom{26}{5}$

7. (a) This is the number of committees of size 7 from a population of size 6, repetition allowed. Stars and bars. Answer is $\binom{7+6-1}{7}$

(b) N(1 one or 2 twos or 3 threes)
= N(1 one) + N(2 twos) + N(3 threes)
− [N(1 one & 2 twos) + N(1 one and 3 threes) + N(2 twos and 3 threes)]
+ N(1 one & 2 twos & 3 threes)
where

N(1 one) = $\binom{6+5-1}{6}$ (put a one on the committee and pick 6 from the other 5 faces)

N(2s twos) = $\binom{5+5-1}{5}$ (put 2 twos on the committee and pick 5 from the other 5 faces)

N(3 threes) = $\binom{4+5-1}{4}$ (pick 4 more from the other 5 faces)

N(1 one & 2 twos)) = $\binom{4+4-1}{4}$ (pick 4 more from the other 4 faces)

N(1 one & 3 threes) = $\binom{3+4-1}{3}$ (pick 3 more from the other 4 faces)

N(2 twos & 3 threes) = $\binom{2+4-1}{2}$ (pick 2 more from the other 4 faces)

N(1 one & 2 twos & 3 threes) = 3 (pick one more from the other 3 faces)

8. N(all C or all T or all A) = N(all C) + N(all T) + N(all A) = $\binom{6}{3} + \binom{7}{3} + \binom{8}{3}$
(the events all C, all T, all A are mutually exclusive so there is nothing to subtract away)

9. *method 1* N(J XOR Q) = N(J and notQ OR Q and notJ)

= N(J and notQ) + N(Q and notJ) (mutually exclusive events)

$= \binom{50}{4} + \binom{50}{4}$

method 2 N(J XOR Q) = N(J OR Q) − N(both) [OR is the usual inclusive or]

$= \underbrace{N(J) + N(Q) - N(both)}_{\text{This is } N(J \text{ or } Q \text{ or both})} - N(both)$

$= \binom{51}{4} + \binom{51}{4} - 2\binom{50}{3}$

method 3 N(J XOR Q) = total − opp

= total − N(neither or both)

= total − [N(neither) + N(both)] (OR rule)

$= \binom{52}{5} - \binom{50}{5} - \binom{50}{3}$

SOLUTIONS Section 1.7

1. (a) total − N(no spades) = $\binom{52}{5} - \binom{39}{5}$

(b) total − N(no spades) − N(1 spade) = $\binom{52}{5} - \binom{39}{5} - 13\binom{39}{4}$

(c) N(4 spades) + N(5 spades) = $\binom{13}{4}\cdot 39 + \binom{13}{5}$

2. N(no spades) + N(1 spade) + N(2 spades) = $\binom{39}{13} + 13\binom{39}{12} + \binom{13}{2}\binom{39}{11}$

3. *method 1*

N(happy Smith family) = N(at least one Smith wins)

= total − N(no Smiths win) = $\binom{100}{3} - \binom{96}{3}$

method 2

N(happy Smith family) = N(1 Smith wins or 2 S's win or 3 S's win or 4 S's win)

= N(1S) + N(2S) + N(3S) + N(4S)

(events are mutually exclusive)

To compute N(1 Smith wins):

Pick the Smith in 4 ways, the other 2 winners in $\binom{96}{2}$ ways.

To compute N(2 Smiths win):

Pick the 2 Smiths in $\binom{4}{2}$ ways, the other winner in 96 ways.

To compute N(3 Smiths win): Pick the 3 Smiths in $\binom{4}{3}$ ways.

Answer is $4\binom{96}{2} + \binom{4}{2}\cdot 96 + \binom{4}{3}$.

method 3

N(happy Smith family) = N(John wins or Mary wins or Bill wins or Henry wins)

= N(John) + N(Mary) + N(Bill) + N(Henry) (4 terms, each is $\binom{99}{2}$)

− [N(JohnMary) + N(JohnBill) + ...] ($\binom{4}{2}$ terms, each is 98)

+ [N(JMB) + N(JMH) + ...] ($\binom{4}{3}$ terms, each is 1)

− N(JMBH) (this term is 0 since there are only 3 winners)

= $4\binom{99}{2} - \binom{4}{2}98 + \binom{4}{3}$

4. No. It double counts. It counts the following outcomes as different when they are really the same (they are both the word ZZABCZ).

outcome 1 Pick spots 1 and 2 for the Z's.
Fill spots 3,4,5,6 with ABCZ.

outcome 2 Pick spots 1 and 6 for the Z's.
Fill spots 2,3,4,5 with ZABC.

Here's the correct version:

N(at least 2 Z's) = total − N(no Z) − N(one Z)

For N(one Z), pick a spot for the Z in 6 ways; then fill each of the other 5 spots with any of the 25 non-Z's. Can be done in $6\cdot 25^5$ ways.

Final answer is $26^6 - 25^6 - 6\cdot 25^5$

5. (a) N(void) = N(at least one missing suit)
= N(no H or no S or no C or no D)
= N(no H) + N(no S) + N(no C) + N(no D)
− [N(no HS) + other.2-at-a-time-terms.]
+ N(no HSC)+ other 3-at-a-time terms

$$= 4\binom{39}{13} - \binom{4}{2}\binom{26}{13} + 4$$

(b) N(S royal flush or H royal flush or D royal flush or C royal flush)
= N(S) + N(H) + N(D) + N(C)− [N(SH)+ other 2-at-a-time terms]
(no room for 3 or more royal flushes)

$$= 4\binom{47}{8} - \binom{4}{2}\binom{42}{3}$$

6. (a) This is about a committee of size 11 from a population of 26.

method 1

N(at most 3 vowels) = N(no vowels or 1 v or 2 v or 3 v)
= N(no vowels) + N(1 v) + N(2 v) + N(3 v)

$$= \binom{21}{11} + 5\binom{21}{10} + \binom{5}{2}\binom{21}{9} + \binom{5}{3}\binom{21}{8} \qquad [\ = 7090496\ ]$$

method 2

N(at most 3 vowels) = total − N(4 vowels or 5 vowels)
= total − N(4 vowels) − N(5 vowels)

$$= \binom{26}{11} - \binom{5}{4}\binom{21}{7} - \binom{21}{6}$$

(b) This is about a committee of size 11 from a population of 26, *repeats allowed* (Stars and Bars, Section 1.5).

method 1

N(at most 3 vowels) = N(no vowels or 1 v or 2 v or 3 v)
N(no vowels) + N(1 vowel) + N(2 vowels) + N(3 vowels)

For N(2 vowels) for instance, pick a subcommittee of 2 vowels (remember that repeats are allowed) and then pick 9 more from the 21 non−vowels. Can be done in $\binom{5+2-1}{2}\binom{21+9-1}{9}$ ways.

Final answer is

$$\binom{21+11-1}{11} + 5\binom{21+10-1}{10} + \binom{5+2-1}{2}\binom{21+9-1}{9} + \binom{5+3-1}{3}\binom{21+8-1}{8}$$

method 2

At most 3 vowels = total − at least 4 vowels
= total − [4 vowels + 5 vowels + 6 vowels + ... + 11 vowels]
(Since repeats are allowed, you can have as many as 11 vowels.)

Total is still $\binom{11+26-1}{11}$

N(4 vowels) = $\binom{5+4-1}{4}\binom{21+7-1}{7}$ (Pick 4 vowels, pick 7 non−vowels)

N(5 vowels) = $\binom{5+5-1}{5}\binom{21+6-1}{6}$

...

N(11 vowels) = $\binom{11+5-1}{11}$

footnote Check that the two methods agree

```
In[5]:=Binomial[21+11−1,11] + 5Binomial[21+10−1,10] +
      Binomial[5+2−1,2]Binomial[21+9−1,9] + Binomial[5+3−1,3]Binomial[21+8−1,8]
Out[5]=493906140

In[6]:=Binomial[11+26−1,11] −  Sum[Binomial[5+n−1,n]Binomial[21+11−n−1,11−n],{n,4,11}]
Out[6]=493906140
```

7.(a) Give each person a cookie. Then toss the other 6 cookies into 4 people.

Answer is $\binom{6+4-1}{6} = 84$ (Stars and bars)

(b) $$\text{total} - \left[\begin{array}{l} N(P_1 \text{ none}) + N(P_2 \text{ none}) + \text{other 1 at a time terms} \\ -\ [N(P_1\&P_2 \text{ none}) + \text{other 2-at-a-time terms}] \\ +\ N(P_1\&P_2\&P_3 \text{ none}) + \text{other 3-at-a-time terms} \end{array}\right]$$

$$= 4^{10} - \left[4.3^{10} - \binom{4}{2}\cdot 2^{10} + 4\cdot 1\right]$$

footnote You can do part (a) using total − opp also:

$$\binom{10+4-1}{10} - \left[4\binom{10+3-1}{10} - \binom{4}{2}\binom{10+2-1}{10} + 4\cdot 1\right] = 84$$

But it's silly because there was a much easier way.
In part (b) there is no easier way.

8. (a) (i) total − N(no b's) $= 3^{10} - 2^{10}$

(ii)
total − N(no a or no b or no c)
= total

$$- \Big[N(\text{no a}) + N(\text{no b}) + N(\text{no c}) - [N(\text{no ab}) + N(\text{no ac}) + N(\text{no bc})] + N(\text{no abc})\Big]$$

$$= 3^{10} - \left[3\cdot 2^{10} - 3\cdot 1 + 0\right]$$

(b) This is a committee-with-repetition-allowed problem (you can't get a committee of 10 from a population of 3 unless you allow repetition)

(i) Put a sure b on the committee. Pick the 9 others. $\binom{9+3-1}{9}$

(ii) Put one of each on the committee. Pick 7 others. $\binom{7+3-1}{7}$

9. N(at least one from H,M,P) = total − N(no H or no M or no P)

$$= \text{total} - \left[\begin{array}{l} \text{no H + no M + no P} \\ -\ (\text{no H \& no M + no H \& no P + no M \& no P}) \\ +\ \text{no H \& no M \& no P} \end{array}\right]$$

$$= \binom{100}{15} - \left[3\binom{98}{15} - 3\binom{96}{15} + \binom{94}{15}\right]$$

10. N(neither cat nor dog) = N(no cat *and* no dog)
= total − N(cat or dog)
= total − [N(cat) + N(dog) − N(cat & dog)]

To count N(contains cat), arrange the 24 objects consisting of the clump CAT and the 23 other letters. Answer is 24!
Similarly N(contains dog) = 24!
To count N(cat & dog) arrange the 22 objects consisting of the clump CAT, the clump DOG and the other 20 letters. Answer is 22!
Final answer is 26! − (24! + 24! − 22!).

11. N(4 pictures and at least one ace)

= N(4 pictures) − N(4 pictures and no aces) $= \binom{16}{4}\cdot 36 - \binom{12}{4}\cdot 36$

12.(a) Pick a spot for A and a spot for B. Fill the remaining spots from 24 letters. Answer is $6\cdot 5\cdot 24^4$.

(b) total − N(no A or no B) = total − [N(no A) + N(no B) − N(no A & no B)]

$$= 26^6 - (25^6 + 25^6 - 24^6)$$

(c) total − N(no A and no B) = $26^6 - 24^6$

(d) N(one B) − N(one B and no A's)

For N(one B) (meaning exactly one B), pick a spot for the B and then fill the other spots with non-B's. Can be done in $6\cdot 25^5$ ways.

For N(one B and no A's), pick a spot for the B and fill the other spots with non-A-non-B's. Can be done in $6\cdot 24^5$ ways.

Answer is $6\cdot 25^5 - 6\cdot 24^5$.

(e) N(two A) − N(two A and no B) − N(two A and one B)

For N(two A), pick 2 spots for the A. Fill the other spots with non-A's.
For N(2 A and no B), pick 2 spots for the A. Fill the other spots with non-A-non-B's.
For N(2 A and 1 B), pick two A spots, pick one B spot, fill the other spots.

Answer is $\binom{6}{2}25^4 - \binom{6}{2}24^4 - \binom{6}{2}\cdot 4\cdot 24^3$

13.(a) Think of the husbands as slots. Fill H_3 in one way, the others in 6! ways. Answer is 6!.

(b) The H_2 and H_5 slots are determined. Fill the others in 5! ways. Answer is 5!

(c) N(H_1or H_2 or ... or H_7 is matched with his wife)

= N(H_1) + ... + N(H_7)

− [N(H_1&H_2) other 2-at-a-time terms)]

+ 3-at-a-time terms − 4-at-a-time terms + 5-at-a-time terms −6-at-a-times

+ N(all match)

There are 7 1-at-a-time terms; by part (a) each is 6!.

There are $\binom{7}{2}$ 2-at-a-time terms; by part (b) each is 5!.

There are $\binom{7}{3}$ 2-at-as-time terms and each is 4!, etc.

$$\text{Answer} = 7\cdot 6! - \binom{7}{2}5! + \binom{7}{3}4! - \binom{7}{4}3! + \binom{7}{5}2! - \binom{7}{6} + 1$$

$$= 7!\ \left(1 - \frac{1}{2!} + \frac{1}{3!} - \frac{1}{4!} + \frac{1}{5!} - \frac{1}{6!} + \frac{1}{7!}\right)$$

(d) total − answer to (c) = 7! − answer to (c)

$$= 7!\left(\frac{1}{2!} - \frac{1}{3!} + \frac{1}{4!} - \frac{1}{5!} + \frac{1}{6!} - \frac{1}{7!}\right)$$

14. *step 1* Pick which 3 colors to have. Can be done in $\binom{5}{3}$ ways.

Say you pick R, W, B.

step 2 Pick a committee of size 30 from a population of size 3 (R, W, B), repetition allowed of course, so as to get at least one of each.

Put one R, one W, one B on the committee to start with and then choose 27 more from R, W, B (stars and bars, Section 1.5).

Equivalently you can think that you have five boxes named R, W, B, G, Y. Toss 30 indistinguishable balls into the boxes so that G and Y are empty and R, W, B are non-empty. Put one ball in each of the R,W,B boxes and toss the other 27 into R,W,B.

Can be done in $\binom{27+3-1}{27}$ ways.

Final answer is $\binom{5}{3}\binom{27+3-1}{27}$.

footnote 1 The 100's are almost irrelevant here. All that matters is that 100 is large enough so that you can pick as many as 28 R (and 1 W and 1 B) or 28 W or 28 B.

footnote 2 You can use total − opp at step 2 as in example 3 but with the stars and bars formula as follows (it takes longer, though):

Total = $\binom{30+3-1}{30}$

Opp = N(no R or no W or no B)

= N(no R) + N(no W) + N(no B) (there are three 1-at-a-time terms)

+ $\Big[$N(no R & no B) + other 2 at a time terms$\Big]$ (there are $\binom{3}{2}$ 2−at-a-time terms)

= $3\binom{30+2-1}{30} - \binom{3}{2}\cdot 1$

Answer is $\binom{5}{3}\left[\binom{30+3-1}{30} - 3\binom{30+2-1}{30} + \binom{3}{2}\cdot 1\right]$

SOLUTIONS Section 1.8

1. (1) $20\cdot19\cdot18$

(2) *method 1* Choose a committee of 3 and then choose one to be the leader. Answer is $\binom{20}{3}\cdot 3$.

method 2 Choose leader. Then choose a committee of two assistants. $20\cdot\binom{19}{2}$

(3) This is a permutation of 20 objects, 3 of which are identical. $\frac{20!}{3!}$

(4) *method 1* total − neither Tom nor Mary $= \binom{20}{6} - \binom{18}{6}$

method 2 N(Tom or Mary) = N(Tom) + N(Mary) − N(both) $= \binom{19}{5} + \binom{19}{5} - \binom{18}{4}$

(5) $\binom{19}{6}$ (choose 6 from the non-Sams)

(6) The books are the slots. 20^5

(7) The books are the slots. $20\cdot19\cdot18\cdot17\cdot16$

(8) Pick a committee of 5 people to get the books. $\binom{20}{5}$

(9) Pick a committee of 5 people but allow repeated members. $\binom{20+5-1}{5}$

2. (a) Pick 3 kings. Pick 2 others. $\binom{4}{3}\binom{48}{2}$

(b) N(3 kings) + N(4 kings) $= \binom{4}{3}\binom{48}{2} + 48$

(c) $\binom{39}{5}$

(d) $\binom{13}{5}$

(e) Pick the suit. Then pick 5 cards from that suit.
Or use
N(all Spades or all H or all C or all D)
= N(all S) + N(all H) + N(all C) + N(all D)

Either way, you get $4\binom{13}{5}$

(f) *method 1*
N(no suit missing) = N(at least one each of C, H, D, S)
= total − N(no C or no H or no D or no S)
= total − [N(no C) + N(no H) + N(no D) + N(no S)
− [N(no C and no H) + other 2-at-a-time terms]
+ 3 at-a-time terms]

$$= \binom{52}{5} - \left[\ 4\binom{39}{5} - \binom{4}{2}\binom{26}{5} + \binom{4}{3}\binom{13}{5}\ \right]$$

method 2 Pick a suit to double up in, pick 2 cards from that suit, pick one card from each of the other two suits. Answer is $4\binom{13}{2}13^3$.

3. $\frac{67!}{33!}$

4. (a) Arrange 12 women and a man-lump. Then permute the lump. $13!\,5!$

(b) Fill the left end, the right end, and then permute the remaining 15 people for the other seats. $5\cdot4\cdot15!$

(c) Line up the 12 women. Then choose places for the men from the 13 between/ends
$12!\cdot13\cdot12\cdot11\cdot10\cdot9$

5. (a) 4^6 (each ball is a slot)

(b) 3^6

(c) total − N(B_2 empty) = $4^6 - 3^6$

(d) N(at least one ball in each box)
= total − N(B_1 empty or B_2 empty or B_3 empty or B_4 empty)

Total = 4^6

N(B1 empty or B2 empty or B3 empty or B4 empty)
= N(B1 empty) + ... + N(B4 empty)
− [N(B1 and B2 empty) + other 2−at−a−time terms] (There are $\binom{4}{2}$ terms here)
+ 3−at−a−time terms
− N(all empty)

$$= 4^6 - \left[4\cdot 3^6 - \binom{4}{2} 2^6 + \binom{4}{3} - 0\right]$$

6.(a) N(no W or no H) = N(no W) + N(no H) − N(no W & no H) = $\binom{17}{12} + \binom{37}{12} - \binom{15}{12}$

(b) N(no W and no H) = N(only non−H men) = $\binom{15}{12}$

7. Choose a committee of size 12 from a population of size 4 with repeats allowed.
Answer is $\binom{12+4-1}{12}$

8. *method 1* Permute the letters. Permute the digits. Then there are 2 ways to put the two arrangements together. Answer is $4!\,4!\cdot 2$

method 2 First spot can be filled in 8 ways, second in 4 ways (must use the other type of symbol), third spot in 3 ways (back to the first type of symbol) etc.
Answer is $8\cdot 4\cdot 3\cdot 3\cdot 2\cdot 2$.

9. (a) N(ace of spades) + N(KQJ spades) − N(AKQJ spades) = $\binom{51}{4} + \binom{49}{2} - 48$

(b) N(all hearts) + N(no hearts) = $\binom{13}{5} + \binom{39}{5}$
(nothing to subtract away since the events are mutually exclusive)

10.(a) *method 1* $10\cdot 9\cdot 8$ (the people are the slots)
method 2 Line up J, M, T and 7 dashes (the identical empty seats) Ans is $10!/7!$

(b) *mehod 1*
Choose 3 adjacent seats. Can be done in 8 ways (S1−S3, S2−S4,..., S8−S10)
Then match the people to the seats. Answer is $8\cdot 3!$
method 2 Line up a people clump and 7 dashes. Then permute within the clump
$$\frac{8!}{7!}\ 3!$$

(c) Permute 3 people and 7 dashes (the empty seats) so that the people are kept apart.

method 1
Line up the 7 dashes (can be done in only one way)
Pick 3 of the 8 between/ends and match them to the people.

Answer is $\binom{8}{3}\ 3!$

Or equivalently pick one of the 8 between/ends for John, one for Mary, one for Tim.
Answer is $8\cdot 7\cdot 6$

method 2 (the first method was easier) (maybe you don't want to read this)
step 1 Line up the people. Can be done in 3! ways
step 2 There are 2 betweens and 2 ends which act like distinguishable boxes.
You want to toss the 7 identical chairs into the 4 boxes so that the 2 between boxes get at least one chair each (to make sure the people are separated)
To do this, toss a chair into each of the two betweens.
Then toss the remaining 5 identical chairs into the 4 boxes.

Can be done in $\binom{5+4-1}{3}$ ways.

Put the two steps together to get answer $3!\binom{5+4-1}{3}$.

11. Choose a committee of 3 to type, a committee of 4 to file and match the remaining 3 with A_1, A_2, A_3 Answer is $\binom{10}{3}\binom{7}{4}\,3!$

12. (a) $10\cdot 20\cdot 30\cdot 40$

(b) $\binom{10}{2}\binom{20}{4}\cdot 30$

(c) total − N(no W) − N(1W) − N(2W) $= \binom{100}{12} - \binom{80}{12} - 20\cdot\binom{80}{11} - \binom{20}{2}\binom{80}{10}$

(d) N(no W) + N(1 W) $= \binom{80}{12} + 20\cdot\binom{80}{11}$

13.(a) Each student is a slot which can be filled in 5 ways. Answer is 5^{29}.

(b) In this context, the 29 students should be treated as indistinguishable balls which are tossed into 5 boxes. Answer is $\binom{29+5-1}{29}$

The students are distinguishable (they do actually have names) but the names don't matter here because we don't care *who* gets A's, just *how many* get A's.

(c) Pick 5 people for the A's, then 6 for the B's etc. $\binom{29}{5}\binom{24}{6}\binom{18}{7}\binom{11}{6}$

14. 2^4 (4 toppings slots, each filled wth yes or no; i.e., this is an all, none, any combination problem)

15. You must walk 4 E's and 5 S's but you can permute the directions any way you like. This amounts to permuting 4 E's and 5 S's. Answer is $\frac{9!}{4!\,5!}$.

16. (a) $\frac{48!}{2!\,46!}\;\frac{3!\,45!}{48!} = \frac{3}{46}$

(b) $\frac{100!}{2!\,98!} = \frac{100\cdot 99}{1\cdot 2} = 50\cdot 99 = 4950.$

17. Count committees (since order doesn't count) of size 5 chosen from a population of 10 with repetition allowed. $\binom{5+10-1}{5} = \binom{14}{5}$

18. *method 1* Pick 4 more letters. Then permute the 5 letters. $\binom{25}{4}\,5!$

method 2 Pick a spot for the B. Then fill the remaining spots. $5\cdot 25\cdot 24\cdot 23\cdot 22$

19.(a) Each day, the cafeteria serves a committee of 3 items. The total number of days in a cycle is the total number of committees which is $\binom{10}{3}$.

(b) Count the number of committees which contain B_3. $\binom{9}{2}$

(c) *method 1* total committees − comms with no B's = $\binom{10}{3} - \binom{7}{3}$

method 2 $N(B_1 \text{ or } B_2 \text{ or } B_3)$

$= N(B_1) + N(B_2) + N(B_3) - [\,N(B_1B_2) + N(B_1B_3) + N(B_2B_3)\,] + N(B_1B_2B_3)$

$= 3\binom{9}{2} - 3\cdot 8 + 1$

(d) N(at least one B and at least one V)

= total menus − N(menus with all B or all V)

= total menus − N(menus with all B) − N(menus with all V)

$= \binom{10}{3} - 1 - \binom{7}{3}$

20. You need a committee with either no negatives or 2 negatives or 4 negatives.

(a) N(no negs) + N(2 negs) + N(4 negs) $= 1 + \binom{5}{2}\binom{4}{2} + \binom{5}{4}$

(b) (stars and bars) $\binom{4+4-1}{4} + \binom{5+2-1}{2}\binom{4+2-1}{2} + \binom{5+4-1}{4}$

21. (a) Arrange 3 U's, N,S,A,L. Answer is $\frac{7!}{3!}$

(b) Permute a U block, N,S,A,L. Amswer is 5!

(c) Permute a vowel block (UUUA) together with N,S,L. Can be done in 4! ways. Then permute the block itself. Can be done in $\frac{4!}{3!}$ ways. Answer is $4!\frac{4!}{3!}$.

(d) Permute N,S,A,L. Then pick 3 of the 5 in-betweens and ends for the U's. $4!\binom{5}{3}$

(e) Permute the vowels U,U,U,A. Can be done in $\frac{4!}{3!}$ ways.
From the 5 in-betweens and ends, pick a spot for N, then a spot for S then for L. Answer is $\frac{4!}{3!}\cdot 5\cdot 4\cdot 3$.

22. The 100 messages are indistinguishable balls being tossed into 4 channels.

(a) $\binom{100+4-1}{100}$

(b) Now we want at least one each in C_1, C_2, C_3 and none in C_4.
Put a ball into each of C_1, C_2, C_3 and toss the rest into 3 boxes. $\binom{97+3-1}{97}$

23. (a) N(1M & 1W or 2M & 2W or 3M & 3W or 4M & 4W)
= N(1M & 1W & 6T) + N(2M & 2W & 4T) + N(3M & 3W & 2T) + N(4M & 4W)
(the events are mutually exclusive so there are no more terms in the OR rule)

$$= 15\cdot 16\binom{17}{6} + \binom{15}{2}\binom{16}{2}\binom{17}{4} + \binom{15}{3}\binom{16}{3}\binom{17}{2} + \binom{15}{4}\binom{16}{4}$$

(b) (see rule for exactlies combined with at leasts)

$$N(2M) - N(2M \,\&\, \text{no } W) - N(2M \,\&\, 1W) = \binom{15}{2}\binom{33}{6} - \binom{15}{2}\binom{17}{6} - \binom{15}{2}\cdot 16\cdot\binom{17}{5}$$

You can also use N(2M & 2W) + N(2M & 3W) + ... + N(2M & 6W).

24. (a) This is like buying 6 doughnuts from a store that sells 26 flavors. The only relevance of the "10 of each letter" is that when the store fills your order it will not run out of any flavor. Choose a committee of 6 from a population of 26, repeats allowed. Answer is $\binom{6+26-1}{6}$.

It doesn't matter whether you draw with or without replacement. All that matters is that each flavor is still available each time you draw. It *would* matter if you were drawing more than 10.

(b) Look at the list of possibilities:
(i) 2A, 2B, 2C
(ii) 2A, 2J, 2Z
etc.

Listing possibilities amounts to picking three letters. Answer is $\binom{26}{3}$.

SOLUTIONS Section 1.9

1. $\binom{49}{6}$

2.
$$\begin{array}{ccccccccccccccc} & & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & & \\ & 1 & & 6 & & 15 & & 20 & & 15 & & 6 & & 1 & \\ 1 & & 7 & & 21 & & 35 & & 35 & & 21 & & 7 & & 1 \end{array}$$

So $(x + y)^7 = x^7 + 7x^6y + 21x^5y^2 + 35x^4y^3 + 35x^3y^4 + 21x^2y^5 + 7xy^6 + y^7$.

Pascal's triangle says that the coeff of x^5y^2 is 21.
By (1), the coeff of x^5y^2 is $\binom{7}{2} = \frac{7!}{2!\,5!} = \frac{7\cdot 6}{2\cdot 1} = 21$. They agree!

3. (a) $\frac{13!}{2!\,4!\,6!}$ (b) $\frac{13!}{10!}$

4. $\frac{13!}{5!\,8!}$

5. You're tossing 17 indistinguishable balls into 4 distinguishable boxes. For example the term $x^2y^3z^4w^8$ corresponds to tossing 2 balls into the x box, 3 into the y box, 4 into the z box, 8 into the w box.

Answer is $\binom{4+17-1}{17} = \binom{20}{17}$.

6. Here's one way to do it.

$$\begin{aligned} \text{Second answer} &= \binom{51}{4} + \binom{51}{4} - 2\binom{50}{3} \\ &= 2\left[\binom{51}{4} - \binom{50}{3}\right] \quad \text{(algebra)} \\ &= 2\cdot\binom{50}{4} \text{ by Pascal's identity which says } \binom{51}{4} - \binom{50}{3} = \binom{50}{4} \\ &= \text{first answer} \end{aligned}$$

SOLUTIONS Section 1.10

1. Use slots Tom, Dick, Harry, Mary $\binom{16}{6}\binom{10}{6}\binom{4}{1}$

2.(a) Answer to new problem where the conversational groups are named G1, G2, G3, G4 is
$\binom{12}{3}\binom{9}{3}\binom{6}{3}$.

Answer to original problem is $\dfrac{\binom{12}{3}\binom{9}{3}\binom{6}{3}}{4!}$

(b) $\binom{12}{3}\binom{9}{3}\binom{6}{3}$ (use the months as slots)

3. (a) If the sections were named S1, S2, S3, S4 then the answer would be
$\binom{100}{25}\binom{75}{25}\binom{50}{25}$. Since the sections are nameless, the answer is $\dfrac{\binom{100}{25}\binom{75}{25}\binom{50}{25}}{4!}$

(b) Use the teachers as slots. $\binom{100}{25}\binom{75}{25}\binom{50}{25}$

(c) There is a John section, a Mary section and two nameless sections.
The John section can be filled in $\binom{98}{24}$ ways.

Then the Mary section can be filled in $\binom{74}{24}$ ways.

And the remaining 50 people can be divided into two nameless sections in $\dfrac{\binom{50}{25}}{2!}$ ways.

Answer is $\binom{98}{24}\binom{74}{24}\dfrac{\binom{50}{25}}{2!}$

(d) *method 1* Total − opp = answer to (a) − answer to (c).
method 2 Pick 23 people to be in the John&Mary section and then split the remaining 75 people into 3 nameless sections.

Answer is $\binom{98}{23}\dfrac{\binom{75}{25}\binom{50}{25}}{3!}$

footnote You can also use total − same section but it's a little messier.
The total is the answer in (a).
To count the ways they can be in the same section, pick 23 more people to join them. Can be done in $\binom{73}{23}$ ways. Then divide the other 50 people into two groups.

Answer is $\dfrac{\binom{75}{25}\binom{50}{25}}{3!} - \binom{73}{23}\dfrac{\binom{50}{25}}{2!}$

4. If the people were divided into 10 named pairs named P1,..., P10 and one bye it could be done in $\binom{21}{2}\binom{19}{2}\ \dots\ \binom{3}{2}$ ways. Answer to the original problem is
$\dfrac{\binom{21}{2}\binom{19}{2}\ \dots\ \binom{3}{2}}{10!}$ which cancels to $\dfrac{21!}{(2!)^{10}\ 10!}$

5. Do (b) first where you can use slots. Pick 9 for the home team, the rest are automatically the aways. Answer to (b) is $\binom{18}{9}$

Then answer to (a) is $\dfrac{\binom{18}{9}}{2!}$

6. First divide the people into two groups G1,G2 of size 5, three groups H1,H2 of size 20, four groups J1,..., J4 of size 10, one red group and one blue group. Can be done in

$$\binom{170}{5}\binom{165}{5}\binom{160}{20}\binom{140}{20}\binom{120}{20}\binom{100}{10}\binom{90}{10}\binom{80}{10}\binom{70}{10}\binom{60}{30}.$$

ways. Divide by 2!3!4! to get the answer to the original problem:

$$\frac{\binom{170}{5}\binom{165}{5}\binom{160}{20}\binom{140}{20}\binom{120}{20}\binom{100}{10}\binom{90}{10}\binom{80}{10}\binom{70}{10}\binom{60}{30}}{2!\ 3!\ 4!}$$

Compact version is $\dfrac{170!}{(5!)^2(20!)^3(10!)^4\ 30!\ 30!\ 2!3!4!}$

7. (a)(b) These are the same problem. In each case you can use slots. In part (a) the slots are named size 50, size 100, size 200. In part (b) the slots can be named A, B, C.

Each answer is $\binom{350}{50}\binom{300}{100}$

(c) together = together in the 50 group or in the 100 group or in the 200 group
To do together in the 50 group, pick 48 people to join J and M, then pick 100 people for the second group. Similarly for the other two terms.

answer $= \binom{348}{48}\binom{300}{100} + \binom{348}{98}\binom{250}{50} + \binom{348}{198}\binom{150}{50}$

8. If the two 5-packs were called red 5-pack and blue 5-pack then you could use red, blue and 10-pack as the three slots and the answer would be $\binom{20}{5}\binom{15}{5}$

Since the two 5-packs are indistinguishable the answer is $\dfrac{\binom{20}{5}\binom{15}{5}}{2!}$

9. (a) Wrong. The answer counts every outcome exactly twice. For instance it counts the following outcomes as different when they are really the same

outcome 1 Pick the 50 people $P_1,\ldots,P_{50}$ with leftovers $P_{51},\ldots,P_{100}$

outcome 2 Pick the 50 people $P_{51},\ldots,P_{100}$ with leftovers $P_1,\ldots,P_{50}$

This method uses a slot named "first group" but indistinguishable groups can't be used as slots.

Fix it up by dividing by 2. The right answer is $\frac{1}{2}\binom{100}{50}$

(b) Right. It's OK to use slots since the two groups are distinguishable from one another----one group is named "the group of size 25" and the other group is named "the group of size 75". This is different from part (a) where the two groups are both size 50 and are indistinguishable.

SOLUTIONS review problems for Chapter 1

1. Remember that $\binom{n}{k} = \binom{n}{n-k}$. So $\binom{r-1}{r-n} = \binom{r-1}{r-1-[r-n]} = \binom{r-1}{n-1}$

2. (a) There are 7 slots and each can be filled in 10 ways. Answer is 10^7.

(b) 9^7 (fill the slots with non-5's)

(c) total − N(no 2's) = = $10^7 - 9^7$

(d) N(at least one 2 or at least one 3) = total − N(no 2's & no 3's) = $10^7 - 8^7$

(e) total − N(no 2's or no 3's) = total − [N(no 2's + N(no 3's) − N(no 2's & no 3's)]
$= 10^7 - (9^7 + 9^7 - 8^7)$

(f) Pick a place for the 3, pick 2 places for the 4's, fill the other places

$7\binom{6}{2}\cdot 8^4$

(g) N(one 3 and at least two 4's)
= N(one 3) − N(one 3 & no 4's) − N(one 3 & one 4)

For one 3, pick a place for the 3, fill the other 6 places with non-3's

For one 3 & no 4's, pick a place for the 3, fill the rest with non-4's, non-3's

For one 3 & one 4, pick a place for the 3, a place for the 4, fill other places

Answer is $7\cdot 9^6 - 7\cdot 8^6 - 7\cdot 6\cdot 8^5$

3. The teams are the slots

(a) A person can't be on more than one team $\binom{50}{5}\binom{45}{4}\binom{41}{9}$

(b) A person can be on more than one team $\binom{50}{5}\binom{50}{4}\binom{50}{9}$

4. (a) Pick one letter to be repeated 4 times. Then you have say 4P's, 1Q, 1R to permute. Answer is $3\cdot\frac{6!}{4!}$

(b) Choose the tripled letter, the doubled letter and then permute. $3\cdot 2\cdot\frac{6!}{3!\,2!}$

5. *method 1* Put A_1 down anywhere. Seats on the circle don't have names (as opposed to seats on a line which can be called left end, second from left etc.) so this first placement has no effect on the overall count. Then seats can be named in relation to A_1, namely 1st seat clockwise from A_1, 2nd seat clockwise from A_1 etc. So now there are 5 slots to fill. Answer is 5!

method 2 Look at the connection between permutations on a line and circular permutations.

Each circular perm can be cut open in any one of 6 places to get a lineup. So number of circ perms × 6 = number of lineups

$$\text{number of circ perms} = \frac{\text{number of lineups}}{6} = \frac{6!}{6} = 5!$$

6. (a) $\binom{20}{9}$ or equivalently $\binom{20}{11}$

(b) The relevant term in the expansion is $\frac{9!}{2!\,3!\,4!}\,a^2(5b)^3c^4$.

Coeff of $a^2b^3c^4$ is $\frac{9!}{2!\,3!\,4!}\cdot 5^3$

7. (a) (i) Among the 45 pairs are P_1P_2, P_1P_3, P_1P_4 etc.

When you pick 5 of these pairs, one possibility is

$P_1P_2,\ P_1P_3,\ P_1P_4,\ P_1P_7,\ P_1P_8$

But this is not a legal outcome in the original problem, i.e., it's not a way to divide 10 people into 5 pairs (because P_1 appears in more than one pair). So the proposed solution is inflated not because it double counts (i.e., counts some legal outcomes more than once) but because it counts some illegals.

(ii) The solution double counts (a lot). For example it counts the following outcomes as different when they are the same:

outcome 1 Pick 5, say P1, P2, P3, P4, P5, leaving P6, P7, P8, P9, P10

Match them as follows

P1	P2	P3	P4	P5
P6	P7	P8	P9	P10

(i.e., P1 is paired with P6, P2 with P7) etc.

outcome 2 Pick 5, say P2, P3, P4, P5, P6, leaving P1, P7, P8, P9, P10.
Match them as follows

P2	P3	P4	P5	P6
P7	P8	P9	P10	P1

(e.g., P2 is matched with P7, P3 with P8 etc)

(iii) The solution uses the pairs as slots so it divides the 10 into a *first* pair, a *second* pair, etc. This solution counts these two outcomes as different when they are really the same:

outcome 1 $P_1P_2,\ P_3P_4,\ P_5P_6,\ P_7P_8,\ P_9P_{10}$

outcome 2 $P_3P_4,\ P_1P_2,\ P_5P_6,\ P_7P_8,\ P_9P_{10}$

(b) $\binom{10}{2}\binom{8}{2}\binom{6}{2}\binom{4}{2}\binom{2}{2}$ divided by 5!

8. N(C_2 gets between 2 and 5)

$= N(C_2 \text{ gets } 2) + N(C_2 \text{ gets } 3) + N(C_2 \text{ gets } 4) + N(C_2 \text{ gets } 5)$

For N(C_2 gets 3) pick 3 of the 11 grants for C_2 and fill the remaining 8 grant slots from the other three companies. Can be done in $\binom{11}{3}3^8$ ways.

Final answer is $\binom{11}{2}3^9 + \binom{11}{3}3^8 + \binom{11}{4}3^7 + \binom{11}{5}3^6$

9. Choose one of A and B to appear first (2 ways)
Fill in three in-betweens ($24\cdot 23\cdot 22$ ways)
Permute the 5-block and the 21 other letters (22! ways)
Answer is $2\cdot 24\cdot 23\cdot 22\cdot 22!$

10. (a) On the one hand,

$$(1+1)^n = 2^n \qquad (\text{since } 1+1=2)$$

On the other hand, by the binomial expansion,

$$(1+1)^n = \binom{n}{0}1^n + \binom{n}{1}1^{n-1}1^n + \dots + \binom{n}{n}1^n = \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{n}$$

So

$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{n} = 2^n$$

(b) Count the number of subsets of a set with n members.
On the one hand,

number of subsets $= 2^n$ (see (3) in Section 1.1)

On the other hand,

number of subsets = number with 0 members
+ number with 1 member
+ number with 2 members
+ ... + number with n members.

$$= \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \ldots + \binom{n}{n}$$

So

$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \ldots + \binom{n}{n} = 2^n \quad \text{QED}$$

11. Toss 10 indistinguishable hours into 3 distinguishable boxes $\binom{10+3-1}{10}$

12. There are three slots, an A course, a B course, a C course $3 \cdot \binom{5}{2} \cdot 4$

13. First count the ways of dividing the offices into three duos D1, D2, D3, one quartet and two quintets Q1, Q2. Can be done in $\binom{20}{2}\binom{18}{2}\binom{16}{2}\binom{14}{4}\binom{10}{5}$ ways.

Answer to the original problem is $\dfrac{\binom{20}{2}\binom{18}{2}\binom{16}{2}\binom{14}{4}\binom{10}{5}}{3!\ 2!}$ (See Section 1.10)

14. First write $\binom{n}{k} + 2\binom{n}{k-1} + \binom{n}{k-2}$ as $\binom{n}{k} + \binom{n}{k-1} + \binom{n}{k-1} + \binom{n}{k-2}$.

Then use Pascal's triangle:

$$\binom{n}{k} + \binom{n}{k-1} = \binom{n+1}{k}$$

$$\binom{n}{k-1} + \binom{n}{k-2} = \binom{n+1}{k-1}$$ {see the first two lines of the triangle below)

And finally $\binom{n+1}{k} + \binom{n+1}{k-1} = \binom{n+2}{k}$ (see the 2nd and 3rd lines in the triangle)

So $\binom{n}{k} + 2\binom{n}{k-1} + \binom{n}{k-2} = \binom{n+2}{k}$

$$\begin{tikzcd}[column sep=tiny]
\binom{n}{k} \arrow[dr] & + & \binom{n}{k-1} \arrow[dl] & + & \binom{n}{k-1} \arrow[dr] & + & \binom{n}{k-2} \arrow[dl] \\
 & \binom{n+1}{k} \arrow[drr] & & + & & \binom{n+1}{k-1} \arrow[dll] & \\
 & & & \binom{n+2}{k} & & &
\end{tikzcd}$$

15. a) N(each activity chosen at least once)
= total − N(no A or no B or no C or no D)

= total − [N(no A) + N(no B) + N(no C) + N(no D)
− [N(no AB) + other 2-at-a-time terms]
+ 3-at-a-time terms
− N(no ABCD)]

$$= 4^7 - 4 \cdot 3^7 + \binom{4}{2} \cdot 2^7 - \binom{4}{3} \cdot 1 + 0$$

(b) *method 1* Pick 4 rest days. Can be done in $\binom{7}{4}$ ways.

Each of the remaining 3 days is a slot.
Can fill the slots with activities in $4\cdot3\cdot2$ ways

Answer is $\binom{7}{4}\ 4\cdot3\cdot2$

method 2 Pick 3 of the 4 activities. Can be done in 4 ways. Say you pick A, C, D
Them permute A, C, D and 4 Rests. Can be done in $7!/4!$ ways.
That tells you what to do on Mon–Sun. Answer is $4\cdot 7!/4!$

16. $$\frac{(2n+3)!}{(2n+1)!\,2!} - \frac{3(n+2)!}{n!\,2!} = \frac{(2n+3)(2n+2)}{2} - \frac{3(n+2)(n+1)}{2}$$
$$= \frac{(n+1)(4n+6-3n-6)}{2} = \frac{n(n+1)}{2}$$

17. N(unfillable orders) = N(orders where A's ≥ 8 or B's ≥ 5 or C's ≥ 3)
= N(A's ≥ 8) + N(B's ≥ 5) + N(C's ≥ 3) − N(B's ≥ 5 & C's ≥ 3)
(forget the other 2-at-a-time terms and the 3-at-a-time terms because they involve committees larger than 8)

Each order is a committee of size 8 chosen from a population of size 3 (namely A, B, C) with repetition allowed.
There's only one committee in which A's ≥ 8, namely the committee with 8 A's.
To count committees with B's ≥ 5, put 5 B's on the committee and choose 3 others from a population of size 3. Can be done in $\binom{3+3-1}{3}$ ways.
To count committees where C's ≥ 3, put 3 C's on the committee and choose 5 more from a population of size 3. Can be done in $\binom{5+3-1}{5}$ ways.
There is only one committee with B's ≥ 5 and C's ≥ 3, namely the committee with 5 B's and 3 C's.
Answer is $1 + \binom{3+3-1}{3} + \binom{5+3-1}{5} - 1 = 31$.

18. *method 1* Pick 3 spots for the 1's. Answer is $\binom{7}{3}$

method 2 Line up three 1's and four 0's. Answer is $\frac{7!}{3!\ 4!}$

19. (a) Use the people as slots. Fill them with chairs. $12\cdot11\cdot10\cdot9\cdot8$

(b) $\frac{12!}{7!}$ (permute 5 distinct objects and 7 identical chairs)

(c) $\binom{12}{5}\cdot 5!$

(d) $5!\ \binom{7+6-1}{7}$

20. There are 17 slots; each can be filled with Yes or No (i.e., you can visit, all, none or any combination). Answer is 2^{17}

21. Fill 4 slots without replacement. $26\cdot25\cdot24\cdot23$

22. (a) Toss indistinguishable marbles into distinguishable boxes $\binom{10+4-1}{10}$

(b) Use the marbles as the slots 4^{10}

(c) Put a marble in each box and then toss the other 6. Answer is $\binom{6+4-1}{6}$

(d) total − N(B_1 gets none or ... or B_{10} gets none)

= total − [N(none in B_1) + N(none in B_2) + N(none in B_3) + N(none in B_4)

− [N(none in B_1B_2) + other 2-at-a-timwe terms]

+ N(none in $B_1B_2B_3$) + other 3-at-a-time terms

− N(none in $B_1B_2B_3B_4$)

$$= 4^{10} - \left[4\cdot 3^{10} - \binom{4}{2}2^{10} + \binom{4}{3}\cdot 1 - 0 \right]$$

warning Can't do part (d) by putting a sure marble in each box as you did in (c) because the marbles here are distinguishable.

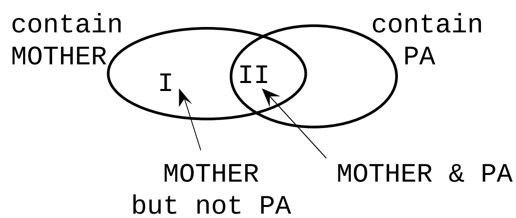
23. (a) The diagram shows that you can find the number of outcomes in region I by taking the entire MOTHER region and subtracting region II. So

N(MOTHER but not PA) = N(MOTHER) − N(MOTHER & PA)

For N(contains MOTHER) line up one MOTHER lump and the other 20 letters. Can be done in 21! ways.

For N(contains MOTHER & PA) line up one MOTHER lump, one PA lump and the other 18 letters. Can be done in 20! ways.

Answer is 21! − 20!



(b) N(neither MOTHER nor PA) = N(no MOTHER and no PA)

= total − N(MOTHER or PA)

= total − [N(MOTHER) + N(PA) − N(MOTHER and PA)]

Total = 26!

For N(MOTHER), permute the MOTHER lump with the 20 other letters.

For N(PA), permute the PA lump with the 24 other letters.

For N(MOTHER and PA), permute a MOTHER lump, a PA lump and the 18 other letters.

Answer is 26! − [21! + 25! − 20!]

24. (a) Pick an academic committee and then an athletic committee $\binom{100}{20}\binom{100}{12}$.

(b) Pick 5 to get both honors, pick 7 of the remaining for athletic honors, 15 of the remaining for scholastic honors. $\binom{100}{5}\binom{95}{7}\binom{88}{15}$

25. (a) Fill 10 slots from the letters x,y,z,w. Answer is 4^{10}.

(b) Pick a committee of 10 from a population of 4 with repetition allowed. $\binom{10+4-1}{10}$

26. Note. A word is a lineup. A card hand is a committee.

(a) *step 1* Pick 5 letters to each appear 3 times.

step 2 Line up the 15 letters (e.g., if you chose the five letters P,Q,A,B,X at step 1 then line up 3 P's, 3 Q's, 3 A's, 3 B's, 3 X's).

Answer is $\binom{26}{5}\dfrac{15!}{3!3!3!3!3!}$.

(b) *step 1* Pick 5 faces.

step 2 Pick 3 cards from each face.

Answer is $\binom{13}{5}\binom{4}{3}\binom{4}{3}\binom{4}{3}\binom{4}{3}\binom{4}{3} = \binom{13}{5}4^5$.

SOLUTIONS Section 2.1

1.

	X	Y	Z
X	1	1	0
Y	1	0	0
Z	0	0	0

2. The row (and column) for that vertex contains all 0's

3. (a) S is even because it's 2E
(b)

$$(*) \qquad S = \deg V_1 + \ldots + \deg V_k + \deg Q_1 + \ldots + \deg Q_n$$

By part (a), S is even.
Each $\deg V_i$ is even so the sum of the $\deg V_i$'s is even.
So (*) is of the form

$$\text{even} = \text{even} + \text{sum of all the Q degrees}$$

So

$$\text{sum of all the Q degrees} = \text{even} - \text{even} = \text{even}$$

Each Q degree is odd. But the *sum* of the Q degrees is even. So there must be an even number of terms in the sum. (you can't add a bunch of odds and get an even sum unless you had an even number of odds in the bunch) (my head is spinning). QED

(c) Let each person be a vertex and draw an edge between any two people who get along. If each person gets along with exactly 5 others then each vertex has degree 5 and the sum of the degrees is 125. But the sum of the degrees has to be even. So it isn't possible for each of the 25 people to get along with exactly 5 others.

4. (a) There are 10 vertices and $\binom{10}{2} = 45$ pairs of vertices. Each pair can be joined by an edge or not joined. So each pair is a slot which can be filled in two ways.
Total number of graphs is 2^{45}.
Note that this pattern works in part (a) also. There were 3 vertices, $\binom{3}{2} = 3$ pairs of vertices and 2^3 different graphs.

(b) Again there are $\binom{10}{2}$ pairs of vertices, each of which can be not joined, joined with one edge or joined with 2 edges. Answer is 3^{45}.
(c) We already know there are 2^{45} ways to draw regular edges.
For each of the 10 vertices there is the further choice of whether to have a loop or not. So there are 2^{10} ways to put in loops.
Final answer is $2^{45} \cdot 2^{10}$

5.(a) Isomorphic.
Pick up vertex P (with edges leading to it intact) and move it down (Fig A).
Then pick up T and move it to the left past Q so that there are no more crossing edges (Fig B)

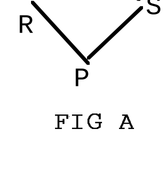


FIG A

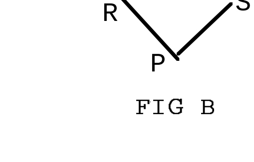


FIG B

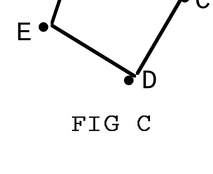


FIG C

Now it looks just like the ABCDE graph (Fig C). Here is a correspondence between the two sets of vertices.

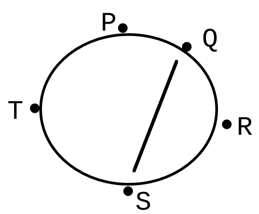
A	T
B	Q
C	S
D	P
E	R

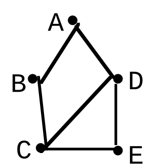
(Could also have A, B, C, D, E corresponding to Q, S, P, R, T or to S, P, R, T, Q, etc.)
Here are the two identical adjacency matrices.

	A	B	C	D	E
A	0	1	0	0	1
B	1	0	1	0	0
C	0	1	0	1	0
D	0	0	1	0	1
E	1	0	0	1	0

	T	Q	S	P	R
T	0	1	0	0	1
Q	1	0	1	0	0
S	0	1	0	1	0
P	0	0	1	0	1
R	1	0	0	1	0

(b) Isomorphic
If you move the edge QS from the "outside" to the "inside" you will see it is just like the first graph.
(Makes no difference whether edges are straight lines or curved lines.)





Here is the correspondence between the two sets of vertices.

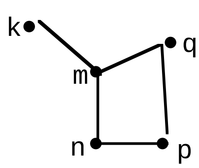
A	P
B	T
C	S
D	Q
E	R

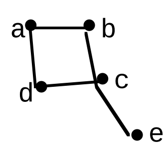
(c) Not isomorphic. In the first graph, S has degree 2 but none of the vertices in the second graph has degree 2.

6. The first and third graphs are isomorphic.
If you move vertex k you can see it. Here are two possible correspondences between the two sets of vertices.

a	p
b	q
c	m
d	n
e	k

a	p
b	n
c	m
d	q
e	k





SOLUTIONS Section 2.2

1. (a)

```
           EBFD
A B C D  ^  A
```

Answer is A B C D EBFD A

(b) D A B C D E B F D (no breakout needed)

```
                RST
(c) P Q R P S Q T  ^  P
```

Answer is P Q R P S Q T RST P

```
                RST
(d) Q P R Q S P T  ^  Q
```

Answer is Q P R Q S P T RST Q

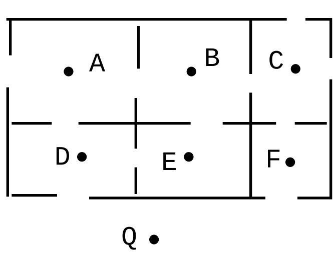
```
(e)    D F H D G
       G  ^  C
     C  ^  E A
 I A  ^  B I
```

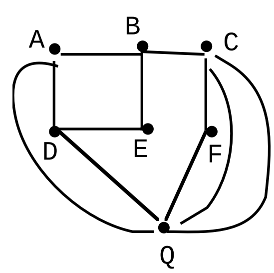
Answer is I A C G DFHDG C E A B I

```
(f)    G H C
     C  ^  D B
 Z A B  ^  A F Z
```

Answer is Z A B C GHC D B A F Z

2. Represent each room by a vertex, let the great outdoors be another vertex, Q, and draw an edge for each door between rooms. Then the problem is to find an Euler cycle (so you can begin and end at Q). But not all vertices have even degree so it's impossible.





3. Not lifting the pencil from the paper and not retracing is equivalent to having an Euler path or cycle. In this case, a path exists since there are two vertices of odd degree (C and D). So it's possible.

Here are the two ways to do it using the breakout algorithm.

Starting at C you get path C B A E B D C E D.

Starting at D you get path D B A E B C D E C

(There are lots of other Euler paths but these are the ones you get by following the breakout algorithm with its alphabetical requirements.)

4. No Euler cycle. There is an Euler path since there are exactly two vertices of odd degree, B and D.

If you start at D and use the breakout algorithm you get $D\ A \overset{FGA}{\wedge} B\ D\ E \overset{HIE}{\wedge} B\ D$.

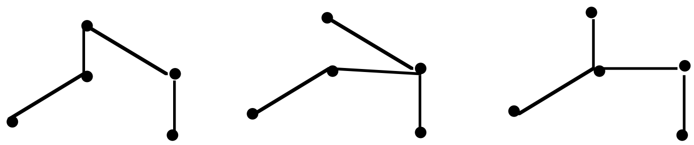
The path is D A F G A B D E H I E B.

If you start at B you get $BA \overset{FGA}{\wedge} D\ B\ E \overset{HIE}{\wedge} D$.

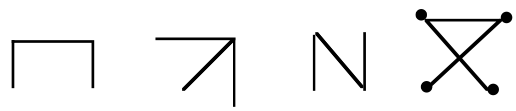
The path is B A F G A D B E H I E D.

SOLUTIONS Section 2.3

1. (a) Must include edges BC and DE. Must have two of the three other edges. There are three spanning trees

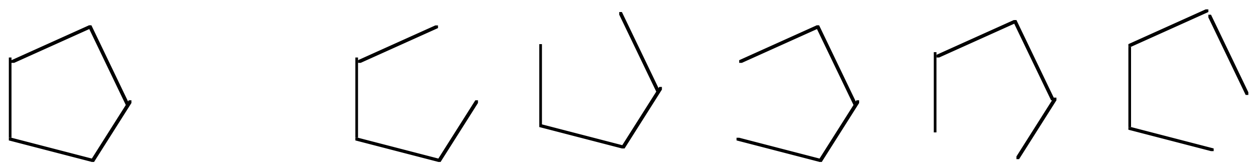


(b) I see 16 spanning trees, four of each of the following types.



2. (a) The only way to get a spanning tree is to delete one of the edges. So there are n spanning trees.

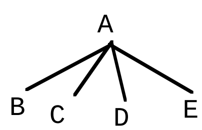
For example here is a cycle with 5 edges and its 5 spanning trees.



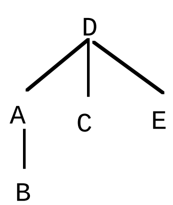
Cycle

Spanning Trees

3.(a)

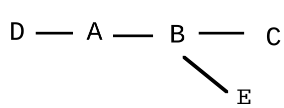


(b)

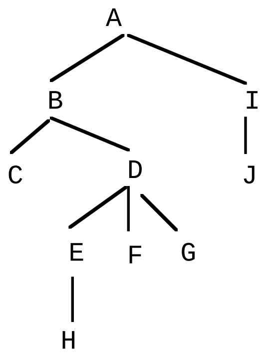


(c) A--B--C--D--E

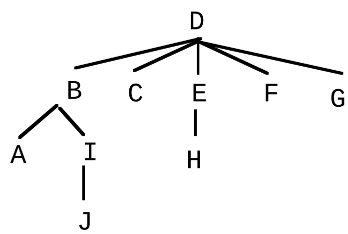
(d)



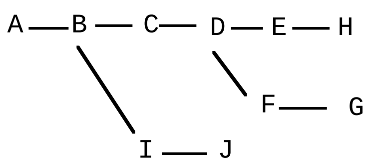
4.(a)



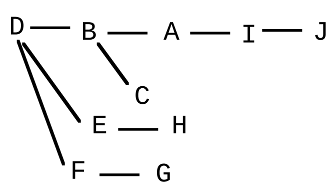
(b)



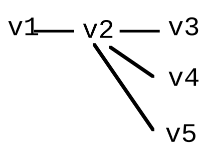
(c)



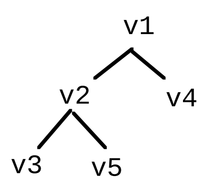
(d)



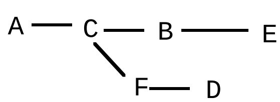
5. (a)



(b)

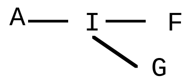


6.(a) Try to get a spanning tree (I started at A and used depth-first).



I did reach all the vertices so the graph is connected.

(b) Try to get a spanning tree (I started at A and used depth-first).
This is as far as I could get so the graph isn't connected.



Try to get a spanning tree for what's left (I started at B).
This is as far as I could go.

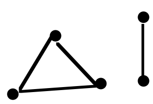
B — D

Try to get a spanning tree for what's left (I started at C).

C — E — J — H

That's all the vertices.
The graph has three components. One component contains vertices A, I, F, G; a second contains B, D and the third component contains the remaining vertices C, E, J, H.

7. (a) Here's a counterexample.



(b) True
If the graph were a tree then V would equal E+1. Contradicting the hypothesis.
So the graph can't be a tree.

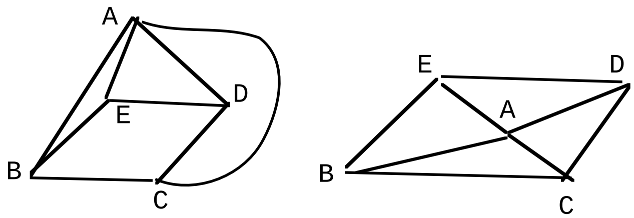
footnote
The statements in (***) is the contrapositive of (*) and contrapositives always have the same truth value (both true or both false).
The statement in (**) is the converse of (*). Converses may or may not have the same truth value.

8. (a) True. G doesn't have property (1) so it can't be a tree.
(b) False. What *is* true is that G is disconnected *or* has a cycle.

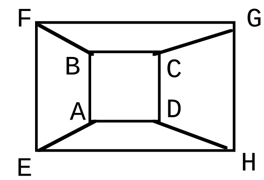
9. Any spanning tree has the same number of vertices as G so it has 13 vertices also. And the tree must have 12 edges by (1).

SOLUTIONS Section 2.4

1. (a) Planar. Don't let the fact that it looks like a picture of a 3-dim tetrahedron fool you. All that matters is that it can be redrawn without edges crossing. The diagram below shows two ways to do it.
(b) Planar. See the diagram for a redrawing.

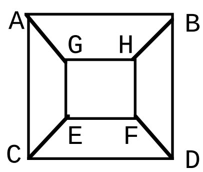


(a)

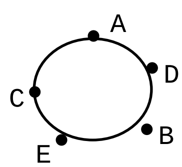


(b)

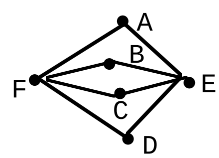
2.



3. See the diagrams below.
(a) $E = 5$, $V = 5$, $F = 2$, $V - E + F = 5 - 5 + 2 = 2$
(b) $E = 8$, $V = 6$, $F = 4$, $V - E + F = 6 - 8 + 4 = 2$

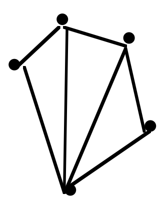


(a)

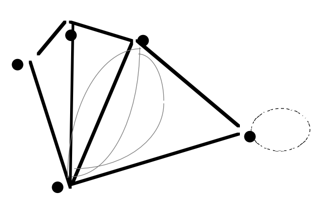


(b)

4. All true.
The diagram illustrates that there is always room to add loops and multiple edges without crossing an existing edge so adding them can't change planar to nonplanar.
And nonplanar-ness is never the fault of loops or multiple edges so deleting them never changes nonplanar to planar.



planar



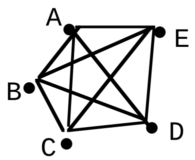
still planar

5. $V = 1000$, $E = 999$, $F = 1$ (the unbounded face, $V - E + F = 1000 - 999 + 1 = 2$

6. Sum of degrees is 28. So $2E = 28$, $E = 14$, $F = 2 + E - V = 7$.

7. Total degrees = $6V$. Also total degrees = $2E$ so $2E = 6V$, $E = 3V$,
$V - E + F = 2$, $V - 3V + 16 = 2$, $V = 7$.

8. Start by drawing the largest cycle possible, ABCDEA (Fig A).
Either BE is inside and AC outside or vice versa. Say BE is inside, AC out (Fig B).
Then BD must be inside and CE must go outside (Fig C)
Now there's no way that edge AD can be drawn without crossing another edge.
So K_5 is non-planar.



K_5

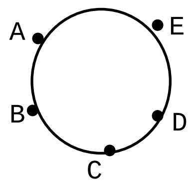


FIG A

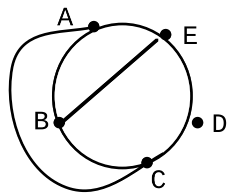


FIG B

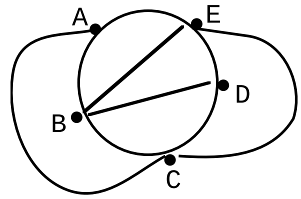
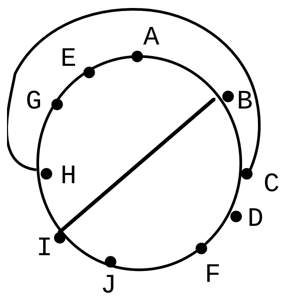
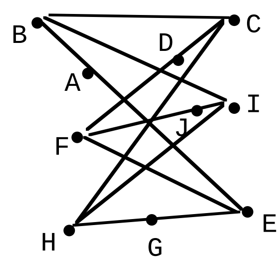


FIG C

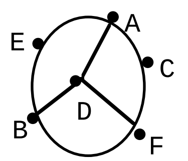
9. (a) Begin say with cycle ABCDFJIHGEA (see lefthand diagram below) If BI goes inside then CH must go outside and then EF can't be drawn without crossing BI or CH. So the graph is nonplanar.

The second diagram shows a subgraph which is a homeomorphic to $K_{3,3}$ (can't contain anything homeomorphic to K_5 since no vertices have degree ≥ 4).

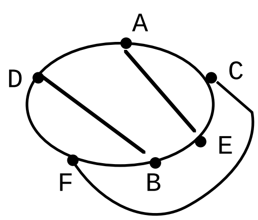




(b) Begin say with cycle ACFBEA.
Vertex D and edges DA, DF, DB are still missing.
Easy to add them without edges crossing. Graph is planar

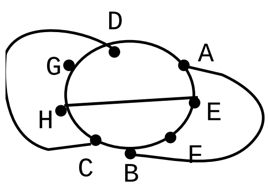


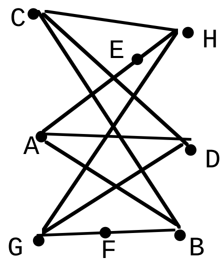
Or you could begin with cycle ACEBFDA. Edges AE, DB and FC are missing. Easy to add without having edges crossing. Graph is planar



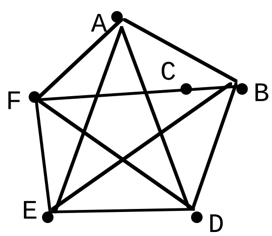
(c) Begin say with cycle AEFBCHGDA. If HE is drawn inside then CD and AB must go outside and GF can't be drawn without crossing. Graph is nonplanar

There are several subgraphs homeomorphic to $K_{3,3}$. The diagram shows one of them.



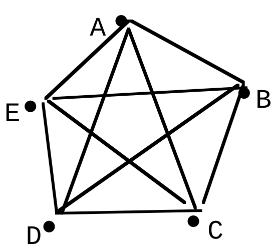


10. There are several subgraphs which are homeomorphic to K_5. The diagram below shows one of them.

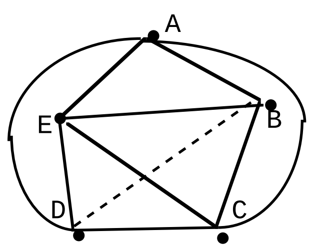


11. (a) Remove an edge. The new graph can't contain $K_{3,3}$ or anything homeomorphic to $K_{3,3}$ since it doesn't even have six vertices. It can't contain anything homeomorphic to K_5 since it no longer has five vertices each at least of degree 4. So it must be planar.

(b) Remove say edge DB in the diagram below and redraw DA and CA outside. Voila, no edges crossing any more. Works similarly if the removed edge is one of the "outside" edges like AB.

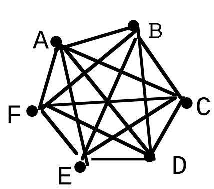


K_5

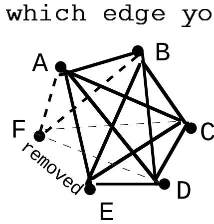


K_5 minus an edge

12. Look at K_6 in the diagram below. Remove say edge EF. The new graph contains K_5 as a subgraph (look at vertices A, B, C, D, E and all edges connecting them) so it's nonplanar. Same thing happens no matter which edge you remove.

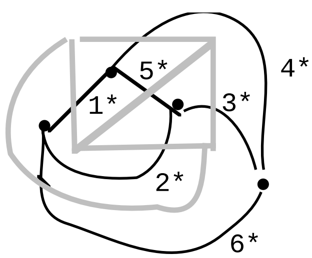


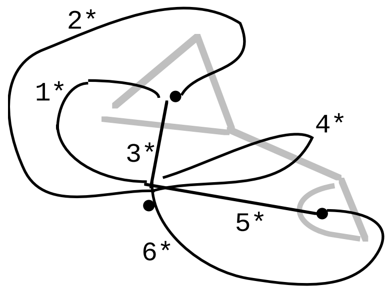
K_6



K_5 subgraph

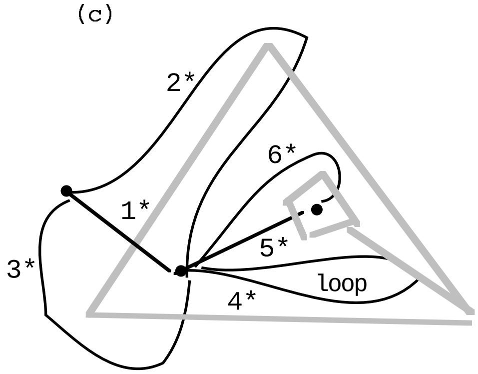
13. (a) (b)



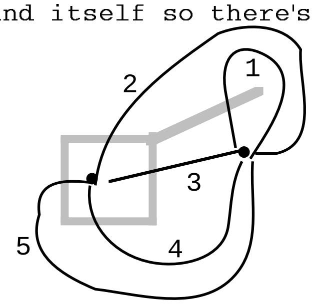


Note that edge 4 in the original is a bdry between the unbounded face and itself so there's a loop 4* in the dual

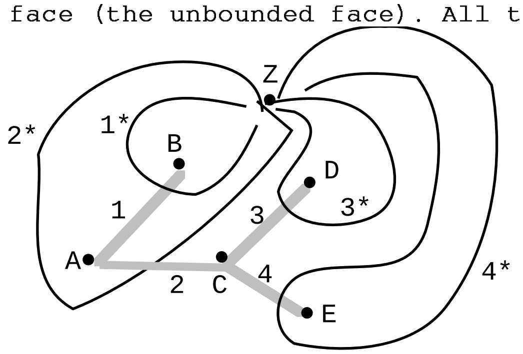
(c)



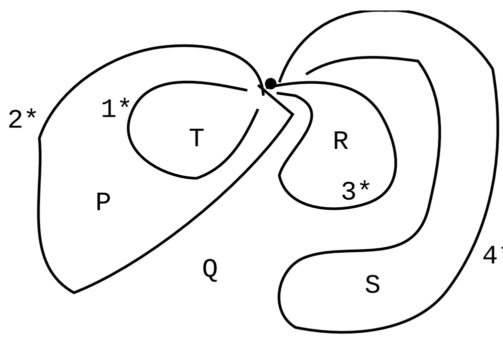
14. Just draw the dual of the given graph. Edge 1* is a boundary between the outer face and itself so there's a corresponding loop in the dual.

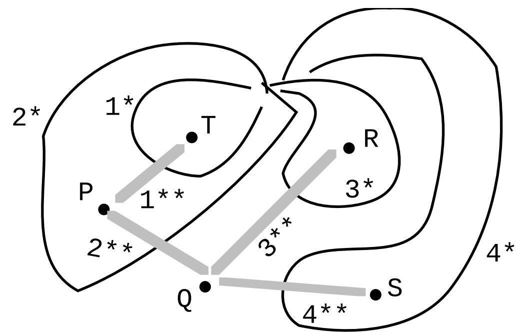


15. (a) The dual has one vertex (I called it Z) because the original graph has only one face (the unbounded face). All the edges in the dual are loops at Z.



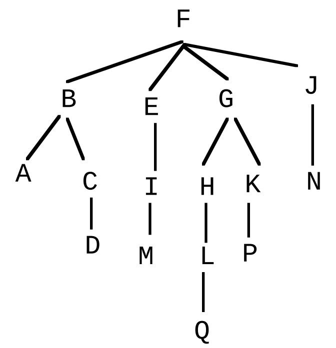
(b) I started with the dual from part (a) (lefthand diagram below) and called the faces P, Q, R, S, T. And then got the dual of the dual (righthand diagram). It *is* isomorphic to the original; the vertices P, Q, R, S, T correspond respectively to A, C, D, E, B.



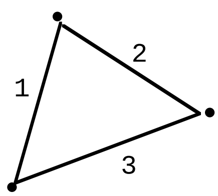


SOLUTIONS review problems for Chapter 2

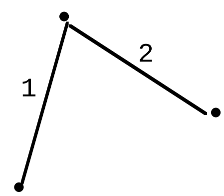
1.



2. False. Here's a counterexample.
The diagram shows a graph and a spanning tree.

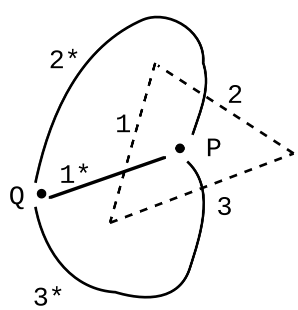


graph

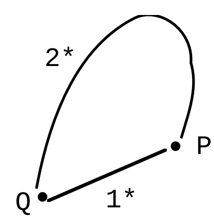


spanning tree

Here is the dual graph (on the left below). The subgraph with edges 1*, 2* is not even a tree, much less a spanning tree for the dual.

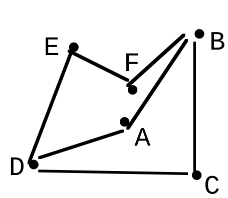


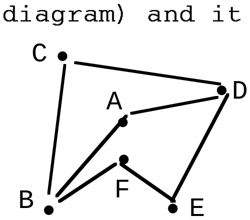
dual graph

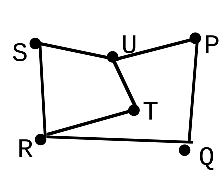


not a spanning tree for the dual

3. Yes
I re-drew the first graph (lefthand diagram below) so that vertex A is *inside* polygon DEFBC and turned it (middle diagram) and it looks just like the PQRSTU graph







Here is the correspondence between the two sets of vertices.

D U
E P
F Q
B R
C S
A T

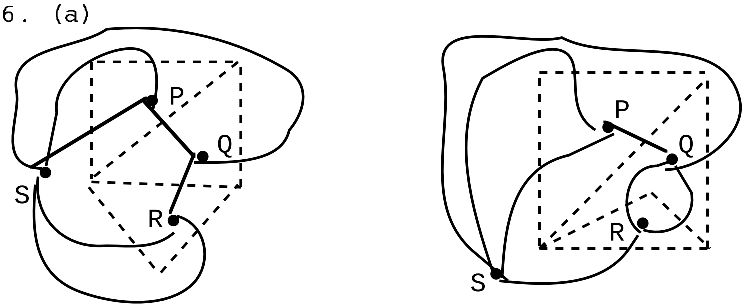
4. There is an Euler cycle since all vertices have even degree. Starting from E, I get

```
             QRB      FGD
      E  A B     C D      E
            ^        ^
```

An Euler cycle is E A B QRB C D FGD E

5. (a) CAN'T. The result has 27 vertices and 25 edges. Since $V \neq E+1$, the result can't even be a tree, much less a spanning tree for G.
(b) MIGHT. The result has 27 vertices and 26 edges. Since $V = E+1$, it could be a tree. If it were a tree than it would be a spanning tree for G since it contains all the vertices of G. But you don't know that it is a tree.
(c) CAN'T. The result might be a tree since $V = E+1$, but even if it were a tree, it wouldn't be a spanning tree for G since it doesn't contain all the vertices of G

6. (a)



(b) The duals aren't isomorphic because the first dual has a vertex, S, with degree 5 and the second dual has no vertex with degree 5.
(c) When you call two graphs isomorphic you're only considering edges and vertices. The two given isomorphic graphs are the "same" from an edge-vertex point of view. But they aren't the same from an edge-*face* point of view: For example, the exterior face of the first graph is bounded by 5 edges but the exterior face of the second graph is bounded by only 4 edges.

The process of finding a dual depends on the edges and the *faces*. So if the two originals are not the same from an edge-*face* point of view, it is not surprising that you can't count on their duals being isomorphic.

7. $E = 16$, sum of degrees $= 4V$. Also sum of degrees $= 2E$ so $4V = 32$, $V = 8$,
By Euler's theorem, $V - E + F = 2$ so $F = 2 + E - V = 10$

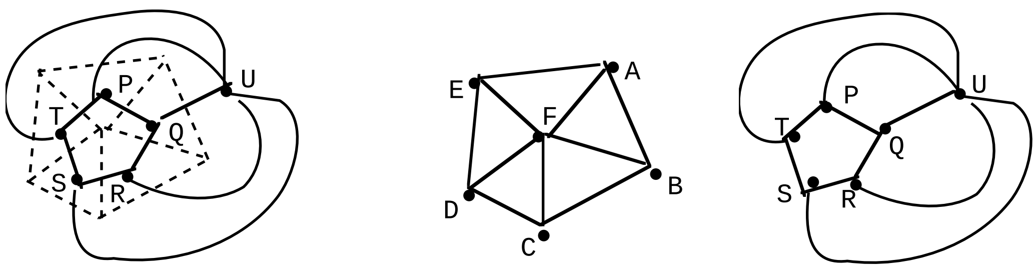
8. Yes.

The lefthand diagram shows the dual, drawn on top of the original.
On the right you can see the original and its dual drawn separately.
If you pick up vertex U (with attached edges) and move it inside polygon PQRST, the dual will look just like the original.

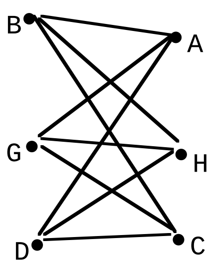
With the correspondence

P	A
Q	B
R	C
S	D
T	E
U	F

the two adjacency matrices are the same.



9. The graph contains several $K_{3,3}$'s. The diagram shows one of them.

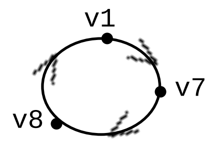


SOLUTIONS Section 3.1

1.(a) No paths to or from v_2. All 0's in row 2 and in col 2 of M_∞

(b) There are paths v_1 to v_7, v_1 to v_8, v_1 to v_1, v_7 to v_8, v_7 to v_1, v_7 to v_7, etc.

	v1	v7	v8
v1	1	1	1
v7	1	1	1
v8	1	1	1



2. There is a path from v_1 to v_2 and from v_2 to v_5 so there is a path from v_1 to v_5 so there must be a 1 in row 1, col 5.

3. M_0 (the adjacency matrix)

There are edges v_7v_1, v_1v_5, v_5v_2, v_2v_3 so there are 1's in

row 7, col 1
row 1, col 5
row 5, col 2
row 2, col 3

M_1

Keep all the 1's from M_0.

Look for paths using v_1 as intermediate.

There is a path $v_7v_1v_5$

So there also is a 1 in row 7, col 5

M_2

Keep all the 1's from M_1.

Look for paths using v_1 and/or v_2 as intermediates.

There is a path $v_5v_2v_3$

So there is also a 1 in row 5, col 3

M_3 and M_4

Just keep the 1's from M_2. No new 1's (that we know of with the given info)

M_5

Keep the 1's from M_4.

Look for paths using $v_1,...,v_5$ as intermediates.

There are paths $v_7v_1v_5v_2$, $v_7v_1v_5v_2v_3$, $v_1v_5v_2$, $v_1v_5v_2v_3$

So there is a 1 in
row 7, col 2
row 7, col 3
row 1, col 2
row 1, col 3

$M_6,...,M_9$

Just keep the 1's from M_5.

(Did I miss any?)

4. (a) There is a path from v1 to v6 using only v1, v2, v3 as possible intermediates.
(b) There is no path from v6 to v1 that goes only through v1, v2, v3 as possible intermediates.

5.(a) *round 1* Look at col 1 in M. There is a 1 in row 5 so add row 1 to row 5.

$$M_1 = \begin{matrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{matrix}$$

round 2 Look at col 2 in M_1. There are 1's in rows 2,4,5. Add row 2 to these rows.

$$M_2 = \begin{matrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{matrix}$$

round 3 Look at col 3 in M_2. There are 1's in rows 1,2,4,5. Add row 3 to them.

$$M_3 = \begin{matrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{matrix}$$

round 4 Look at col 4 in M_3. There are 1's in every row. Add row 4 to every row.

$$M_4 = \begin{matrix} 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{matrix}$$

round 5 Look at col 5 in M_5. There are 1's in every row so add row 5 to every row.

$M_\infty = M_5$ = all 1's

So there are paths anywhere you want to go

(b) $M_1 = \begin{matrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{matrix}$, $M_2 = \begin{matrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{matrix}$, $M_\infty = M_3 = M_2$

6. (a) $R_1 = \begin{matrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{matrix}$, $R_2 = R_1$, $R_\infty = R_3 = \begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$

(b) $R_1 = \begin{matrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{matrix}$, $R_2 = \begin{matrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{matrix}$, $R_3 = \begin{matrix} 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{matrix}$

$R_4 = R_3$, $R_\infty = R_5 = R_4$

(c) $R_1 = R$, $R_2 = R_1$, $R_3 = R_2$, $R_4 = \begin{matrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{matrix}$, $R_\infty = R_5 = R_4$

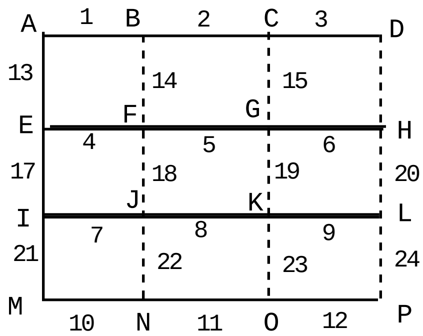
6. $R_1 = \begin{matrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$ No changes from now on. $R_\infty = R_4 = R_1$

(a) yes because there's a 1 in row 2 col 4 of R_∞

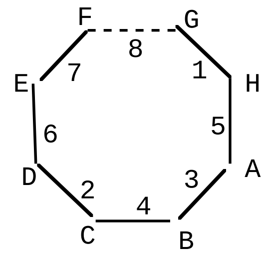
(b) no because there's a 0 in row 4, col 2

SOLUTIONS Section 3.2

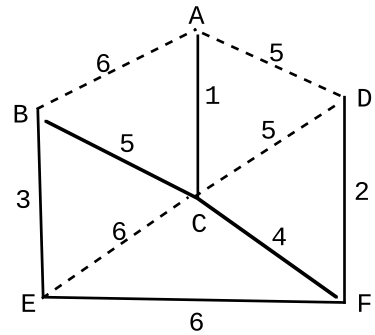
1. (a) Choose edges AB, BC, CD, AE, EF, FG, GH, EI, IJ, JK, KL, IM, MN, NO, OP
 (b) AB, BC, CD, AH, HG, DE, EF (obvious that a spanning tree must omit one edge and to get the min spanning tree it should omit the heaviest edge FG)
 (d) AC, CF, DF, BC, EB



(a)



(b)



(c)

2.(a)

	A	B	C	D	E	F
A	0	6	1	5	∞	∞
B	6	0	5	∞	3	∞
C	1	5	0	5	6	4
D	5	∞	5	0	∞	2
E	∞	3	6	∞	0	6
F	∞	∞	4	2	6	0

(b) I'll start the tree with vertex A

	B	C	D	E	F
status	x	x	x	x	x
link	A	A	A	A	A
cost	6	1	5	∞	∞
status	x	√	x	x	x
link	C	A	A	C	C
cost	5	1	5	6	4
status	x	√	x	x	√
link	C	A	F	C	C
cost	5	1	2	6	4
status	x	√	√	x	√
link	C	A	F	C	C
cost	5	1	2	6	4
status	√	√	√	x	√
link	C	A	F	B	C
cost	5	1	2	3	4

Now pick E and you're finished. Edges BC, CA, DF, EB, FC are a min spanning tree.
Cost of the tree is 5 + 1 + 2 + 3 + 4 = 15.

3. Start with vertex A.

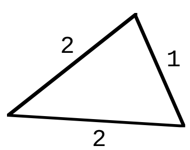
	B	C	D	E
status	x	x	x	x
link	A	A	A	A
cost	2	∞	∞	3

Pick B.
Update C

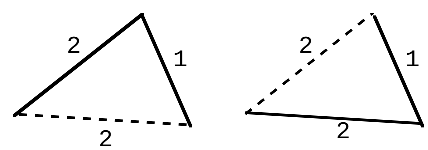
status	√	x	x	x
link	A	B	A	A
cost	2	1	∞	3
status	√	√	x	x
link	A	B	C	A
cost	2	1	4	3
status	√	√	x	√
link	A	B	C	A
cost	2	1	4	3

Pick D and you're finished. A min spanning tree has edges AB, CB, DC, EA
Its weight is 2 + 1 + 4 + 3 = 10.

4. Here's one possibility



weighted graph



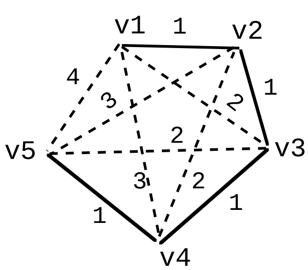
two minimal spanning trees

5.(a) For G_5, the spanning tree has edges v_1v_2, v_2v_3, v_3v_4, v_4v_5. Total weight is 4.
For G_n in general, the edges $v_1v_2, v_2v_3, \ldots, v_{n-1}v_n$ are the cheapest. The min spanning tree has $n-1$ edges each with weight 1 so its total weight is $n-1$

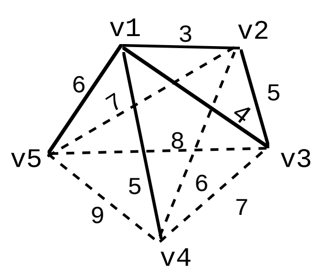
(b) For G_5, edges v_1v_2, v_1v_3, v_1v_4, v_1v_5 form a min spanning tree.
Total weight is 3 + 4 + 5 + 6.
For G_n in general, the edges $v_1v_2, v_1v_3, \ldots, v_1v_n$ are a min spanning tree.

$$\text{Total weight is } 3 + 4 + 5 + \ldots + n+1 = \frac{(n+1)(n+2)}{2} - (1 + 2) = \frac{n^2 + 3n - 4}{2}$$

For reference: $1 + 2 + 3 + \ldots + n = \dfrac{n(n+1)}{2}$

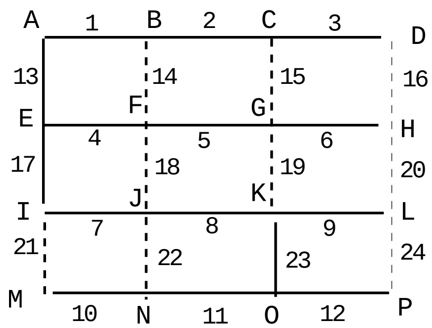


(a)



(b)

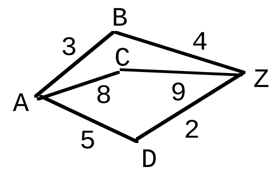
6.(a) Begin with the required edge KO. Then swing into the greedy routine and choose KJ, JI, KL, ON, NM, OP, IE, EF, FG, GH, EA, AB, BC, CD (see the diagram).



(b) The min spanning tree from #1(a) has weight 129. The tree from part (a) here has weight 133. So this tree is not a min spanning tree. It's minimal with respect to all the spanning trees containing edge KO but it isn't minimal overall.

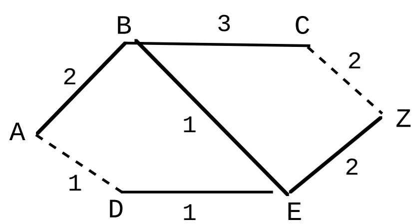
SOLUTIONS Section 3.3

1. Here's a graph with three paths from A to Z and two of them, ABZ and ADZ, are equally shortest.

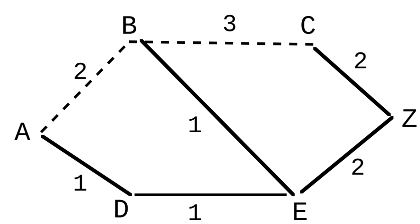


2. (a) P is unpicked. Of all the paths between Q and P that go through only D or F or X or Z (the picked vertices), the shortest has length 8 (could get a shorter path later in the algorithm as other vertices get picked).
(b) X is a picked vertex. Of all paths between Q and X, *the* shortest has length 7. Final result.(That path happens to go through only D or F or Z, the picked vertices, but it can't be improved upon as more vertices are picked.)

3. (a) Start with B.
Pick vertex E, edge BE.
Next comes a tie with A and path BA vs. D and path BED. Pick say A and edge AB.
Pick D and ED.
Pick Z and EZ (see path BEZ). Finally pick C and BC
From the tree below you can read these shortest paths.
Shortest path B to A is BA, length is 2.
Shortest path B to C is BC, length is 3.
Shortest path B to D is BED, length is 2.
Shortest path B to E is BE, length is 1.
Shortest path B to Z is BEZ, length is 3.
(b) Start with B. Then pick E and edge BE, D and DE, A and AD, Z and EZ, C and CZ.
Total weight is 7



Problem 3(a)



Problem 3(b)

4.

vertex	B	C	D	E	Z
status	x	x	x	x	x
link	A	A	A	A	A
cost	2	∞	1	∞	∞
	x	x	√	x	x
	A	A	A	D	A
	2	∞	1	2	∞
	√	x	√	x	x
	A	B	A	D	A
	2	5	1	2	∞
	√	x	√	√	x
	A	B	A	D	E
	2	5	1	2	4
	√	x	√	√	√
	A	B	A	D	E
	2	5	1	2	4

Pick D and update.
dist D+DB < dist B? $\infty \le 2$? No. No change.
dist D+DC < dist C? $\infty < \infty$? No. No change.
dist D+E < dist E? $2 < \infty$?
Set dist E =2, link E = D.
dist D+Z < dist Z? $2 < \infty$? No. No change.

Pick B and update.
dist B + BC < dist C? $5 < \infty$? Yes. Change.
dist B+BE < dist E? 3 < 2? No. No change
dist B+BZ < dist Z? $\infty < \infty$? no change

Pick E and update.
dist E+EC < dist C? $\infty < 5$? No. No change.
dist E+EZ < dist Z? $4 < \infty$? Change.

Pick Z and update.
dist Z+ZC < dist C? 6 < 5? No. No change.
Pick C and you're finished.

Shortest path A to B is AB, length = dist B = 2
Shortest path A to C is ABC, length = dist C = 5
Shortest path A to D is AD, length = dist D = 1
Shortest path A to E is ADE, length = dist E = 2
Shortest path A to Z is ADEZ, length = dist Z = 4

5.(a)

vertex	A	B	D	E	F	G
status	x	x	x	x	x	x
link	C	C	C	C	C	C
cost	6	2	1	4	2	∞

Pick D and update.
dist D + DA = ∞, no change
dist D + DB = 5, dist B = 2, no change
dist D + DE = 3, dist E = 4, change
dist D + DF = ∞, no change
dist D + DG = 5, dist G = ∞, change

vertex	A	B	D	E	F	G
status	x	x	√	x	x	x
link	C	C	C	D	C	D
cost	6	2	1	3	2	5

Tie between B and F. Pick say B.
dist B + BA = 5, dist A = 6, change
dist B + BE = ∞, no change
dist B + BF = ∞, no change
dist B + BG = ∞, no change

vertex	A	B	D	E	F	G
status	x	√	√	x	x	x
link	B	C	C	D	C	D
cost	5	2	1	3	2	5

Pick F and update.
dist F + FA = ∞, no change
dist F + FE = 4, dist E = 3, no change
dist F + FG = 6 , dist G = 5, no change

vertex	A	B	D	E	F	G
status	x	√	√	x	√	x
link	B	C	C	D	C	D
cost	5	2	1	3	2	5

Pick E and stop. No need to update.
Shortest distance between C and E is dist E = 3.
A shortest path is CDE.

(b)

vertex	A	C	D	E	F	G
status	x	x	x	x	x	x
link	B	B	B	B	B	B
cost	3	2	4	∞	∞	∞
status	x	√	x	x	x	x
link	B	B	C	C	C	B
cost	3	2	3	6	4	∞
status	√	√	x	x	x	x
link	B	B	C	C	C	B
cost	3	2	3	6	4	∞
status	√	√	√	x	x	x
link	B	B	C	D	C	D
cost	3	2	3	5	4	7
status	√	√	√	x	√	x
link	B	B	C	D	C	D
cost	3	2	3	5	4	7
status	√	√	√	√	√	x
link	B	B	C	D	C	E
cost	3	2	3	5	4	6

Pick G and you're finished

Shortest path between A and B is AB, length is dist A = 3.
Shortest path between C and B is BC, length is dist C = 2.
Shortest path between D and B is BCD, length is dist D = 3.
Shortest path between E and B is BCDE, length is dist E = 5.
Shortest path between F and B is BCF, length is dist F = 4.
Shortest path between G and B is BCDEG, length is dist G = 6.

6.

B	C	D	E	F	G	Z
x	x	x	x	x	x	x
A	A	A	A	A	A	A
2	∞	∞	∞	1	∞	∞
x	x	x	x	√	x	x
A	A	F	A	A	F	A
2	∞	4	∞	1	6	∞
√	x	x	x	√	x	x
A	B	F	B	A	F	A
2	4	4	6	1	6	∞
√	√	x	x	√	x	x
A	B	F	B	A	F	C
2	4	4	6	1	6	5

Pick D and stop.
A shortest path between A and D is AFD. Its length is dist D = 4

7. (a) True. Say B is picked on round 10. Vertex Z has not been picked yet so *at that moment* dist B ≤ dist Z. Once B is picked, dist B never changes. As you update after picking B, either dist Z remains the same or it changes to dist B + BZ. In either case dist B ≤ dist Z. And Z is picked next so dist Z never changes again. So at the end of the algorithm dist B is still ≤ dist Z.
(b) Yes. By part (a), in the final table, dist B ≤ dist Q and dist Q ≤ dist Z so dist B ≤ dist Z.

8. (a) G has not been picked yet.
The shortest distance between B and G *using picked vertices* (A, C, D, F) *as the only possible intermediates* is 7. The shortest path using picked vertices as the only possible intermediates is GDCB
(There may be a better path and dist when more vertices get picked.)
(b) F is picked.
Shortest distance between B and F is 4. The shortest path itself is FCB.
This is a final result (will never improve as more vertices get picked).

SOLUTIONS Section 3.4

1. Round 1 takes n−1 comparisons.
Round 2, with only n−1 numbers to look at, takes n−2 comparisons etc.
Total number of comparisons is

$$(n-1) + (n-2) + \ldots + 1 = \frac{(n-1)n}{2}$$

So the running time is n^2.

2. (a) You start by going through the (inner) j-loop n times for a total of n comparisons.
You go through the (middle)} i-loop n times and through the (outer) k-loop n times. Final total is $n \cdot n \cdot n = n^3$ comparisons. Running time is n^3.

(b) Each comparison is preceded by an addition so each time you go around the inner loop there are 2n operations. Total number of operations is $2n \cdot n \cdot n = 2n^3$. Running time still is n^3 (same answer whether or not you count the additions).

(c) You go around a first loop n times.
Then you go around a second loop n times.
Then you go around a third loop n times.

There are n + n + n = 3n comparisons. (The number of comparisons is n PLUS n PLUS n, not $n \cdot n \cdot n$, because no one loop is contained in another.)

So the running time is n.

3. Use the comparison as the unit of measure. There are n−1 candidates for the first choice of nearest neighbor and it takes n−2 comparisons to pick the smallest of n−1 edges. At the next round there are n−2 candidates for nearest neighbor and it takes n−3 comparisons to make the choice, and so on. Total number of comps is

$$(n-2) + (n-3) + \ldots + 1 = \frac{(n-2)(n-1)}{2}$$

Running time is n^2.

4. The plain Dijkstra should be run n times, once for each of the vertices serving as the fixed vertex. Each running takes n^2 comparisons (see example 3) so over-and-over requires $n \cdot n^2$ comparisons. Running time is n^3.

5. (a) Yes.
Remember that John is saying that the running time is $\le n^2$ and Mary is saying that it's $\le n^3$.

If the actual running time is $\le n^2$ so that John is right then Mary is automatically right too since anything that is $\le n^2$ is also $\le n^3$. In this case they would both be right but John is more informative.
So on an exam if you answer $O(n^5)$ and I answer $O(n^4)$ then you're right but I won't like your answer and won't give it much, if any, credit .

(b) Yes If some problems of size n actually take n^3 running time and some take less than n^3 then the algorithm's running time is $O(n^3)$ but not $O(n^2)$.
(c) No Once the running time is $O(n^2)$ it is also O(anything larger than n^2) but the latter is less informative.

6. I'll use the comparison "is it positive" as the unit of measure.

In round 1, at worst, there are n comparisons going across row 1 and then there are n more going down col 1. The total is 2n comparisons (*not* n^2). (It's an "at worst" because you might get an early "no" in which case the algorithm jumps to round 2.)

The algorithm does this for n rounds so at worst there are $2n \cdot n = 2n^2$ comparisons.

The running time is $O(n^2)$.

(Use the big oh notation because it was an "at worst" estimate.)

7.(a) number of seconds = n^2 operations $\times 10^{-6}$ secs/op

$n^2 = 60 \cdot 10^6$

$n = \sqrt{60 \cdot 10^6} \approx 7745.97$ so you can do a problem with n = 7745.

(b) $n = \sqrt[3]{60 \cdot 10^6} \approx 391.5$ so you can do a problem with n = 391.

8. Let's say that each operation takes one "time unit".
With the n^3 algorithm and faster computer the problem takes $\frac{1}{10} n^3$ time units
With the n^2 algorithm and old computer the problem take n^2 time units.

So it boils down to which is smaller, $\frac{1}{10} n^3$ or n^2.

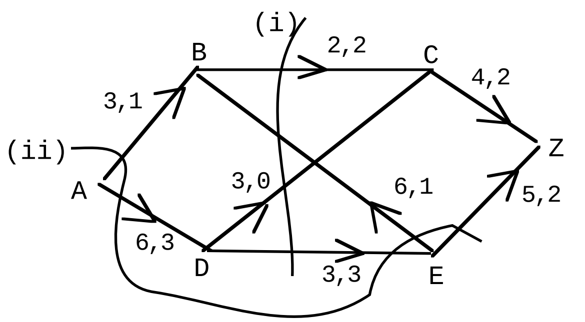
		$\frac{1}{10} n^3$	n^2	
(a)	n=1000	10^8	10^6	use n^2 algorithm
(b)	n=10	10^2	10^2	makes no difference
(c)	n=9	72.9	81	use faster computer

SOLUTIONS Section 3.5

1. (a) By conservation of flow, out of C = into C, x + 1 = 4, x = 3,
Into B = out of B, x + 3 = y, y = 6 (Also into Z = out of A so y+1 = 7, y=6.)
(b) Capacity = BZ cap + AC cap = 5 + 9 = 14, flow = 4 + y − x = 7
(c) 7 since all cuts carry the same flow.

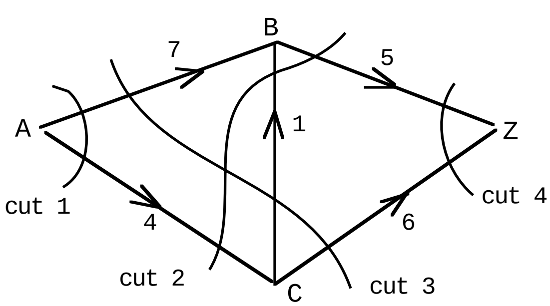
2. NO. The cut is A,B versus C,Z. Edge CB goes from the Z side to the A side and it must carry 0 gallons for the cut to be saturated.

3. (a) Edge EB goes from the Z side to the A side so it does not contribute to the cut capacity and contributes negatively to the cut flow.
capacity = BC + DC + DE = 2 + 3 + 3 = 8
flow = BC + DC + DE − EB = 2 + 0 + 3 − 1 = 4
(ii) Edge EB goes from the A side to the Z side so it contributes to the capacity and to the flow. Similarly for EZ, AB, AD.
But edge DE goes from the Z side to the A side; it doesn't contribute to the capacity and contributes negatively to the flow.
capacity = AB + AD + EZ + EB = 3 + 6 + 5 + 6 = 20
flow = AB + AD + EZ + EB − DE = 1 + 3 + 1 + 2 − 3 = 4

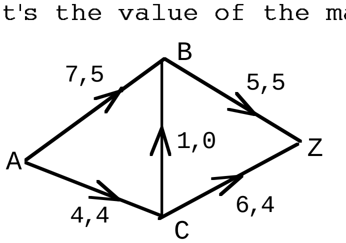


(b) A cut automatically has A on one side and Z on the other. Each of the remaining vertices (B, C, D, E) has to choose to be either on the A side or on the Z side. So there are four slots to fill, each can be filled in 2 ways (choose A or choose Z). Answer is 2^4.
This is also the same as picking all, none or any combination of B,C,D,E.
Also the same as all subsets of {B, C, D, E}

4. There are 4 cuts.



cut 1 A vs. B, C, Z capacity = AB + AC = 11
cut 2 A, B vs. C, Z capacity = AC + BZ = 9
cut 3 A, C vs. B, Z capacity = AB + CB + CZ = 14
cut 4 A, B, C vs. Z capacity = CZ + BZ = 11
Minimum cut capacity is 9 (from cut 2)
That's the value of the max flow. Here's an actual max flow.

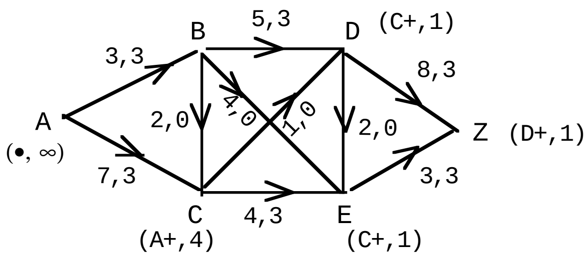


5.	C $(P^+,3)$	D $(P^+,3)$	E no label	F $(P^+,1)$	
	G $(P^+,2)$	H $(P^-,4)$	I no label	J $(P^-,2)$	
	K $(A^+,1)$	L $(P^+,1)$	M $(P^-,1)$	N $(P^-,2)$	Q $(P^-,1)$

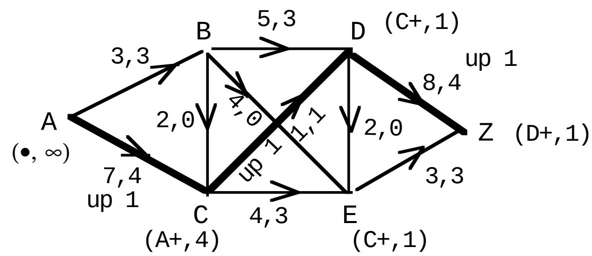
6. No. A vertex can get a neg label only if an edge leads *from* it *to* a labeled vertex and no edges ever lead from Z (since Z is a sink).

7. Z can't get the label (E+, 2) because E doesn't have a label. Can't send another 2 gallons from E to Z because E can't get that extra 2 gallons from somewhere else.

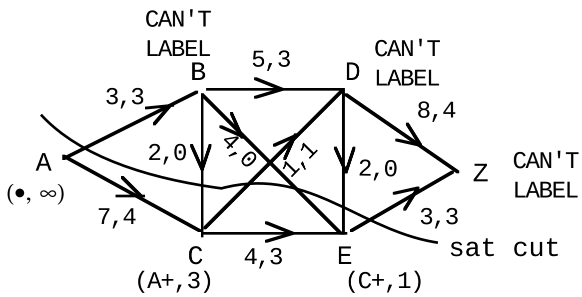
8.(a) *round 1* Label a path to Z



Augment along that path

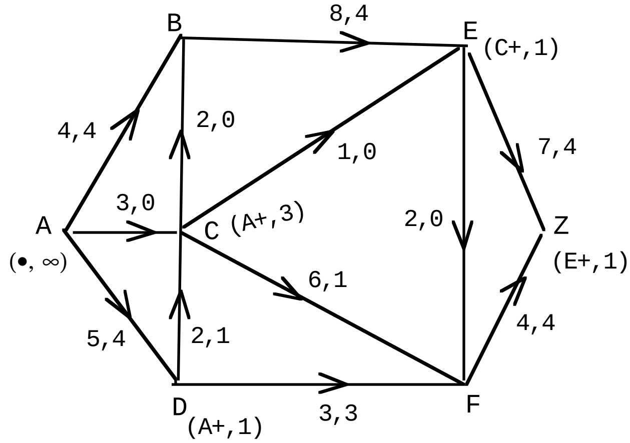


round 2 Try to label a path to Z

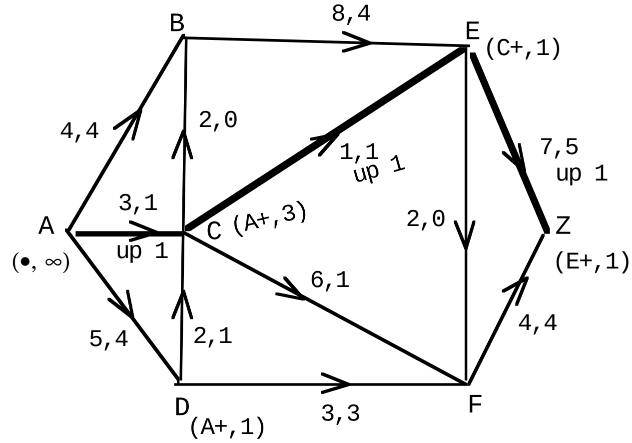


The cut separating the labeled vertices A, C, E from the can't-be-labeled B, D, Z is a saturated cut.
Max flow = sat cut flow = 3 + 1 + 3 = 7 (same as into Z, same as out of A)

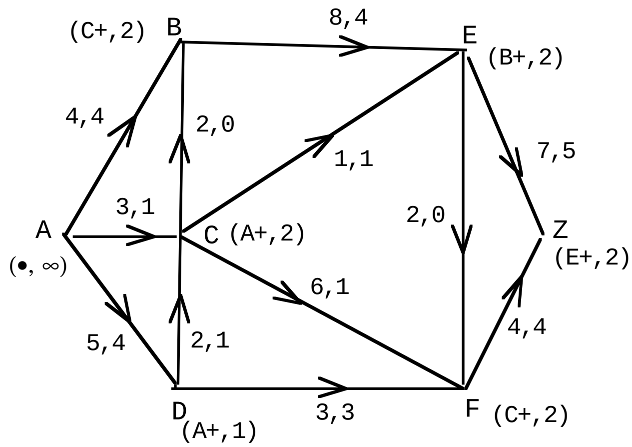
(b) *round 1* Label a path to Z



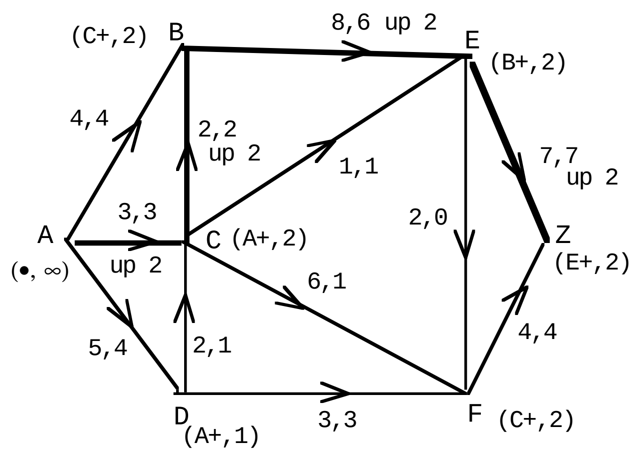
Augment along that path



round 2 Label a path to Z

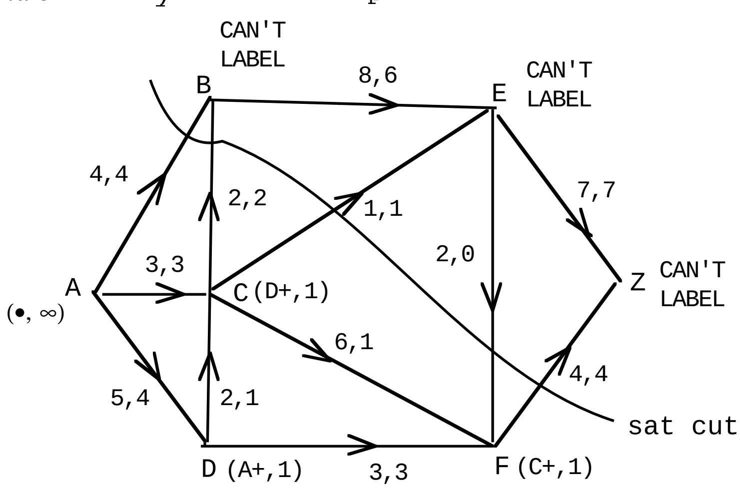


Augment along that path



warning At this stage you can see a saturated cut consisting of edges EZ, FZ by inspection. But the algorithm isn't over until the next round when the labeling gets stuck. You can't stop here on exams.

round 3 Try to label a path to Z

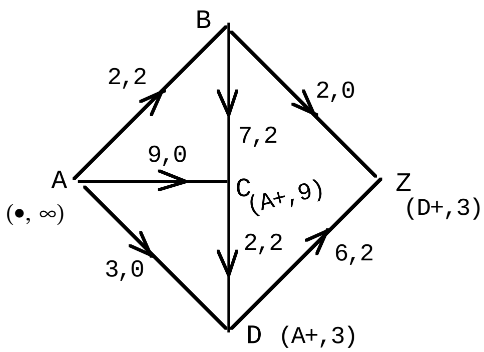


The cut separating the labeled vertices A,C,D,F from the can't-be-labeled B,E,Z is a saturated cut.

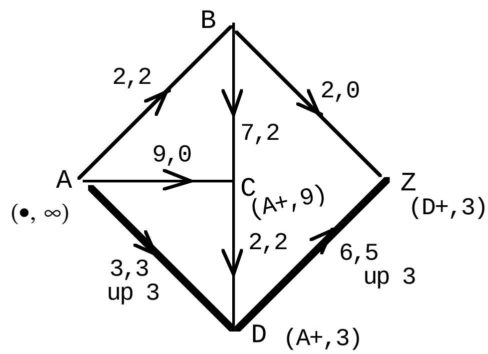
Max flow = flow on sat cut = 4 + 2 + 1 + 4 = 11

(same as into Z, same as out of A)

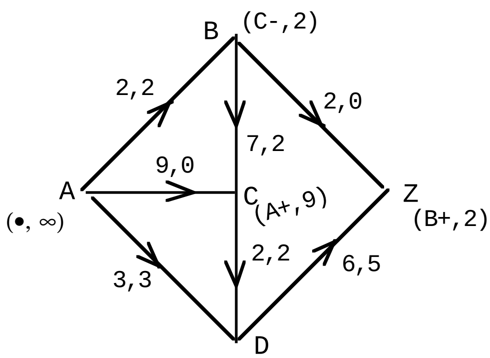
(c) *round 1* Label a path to Z



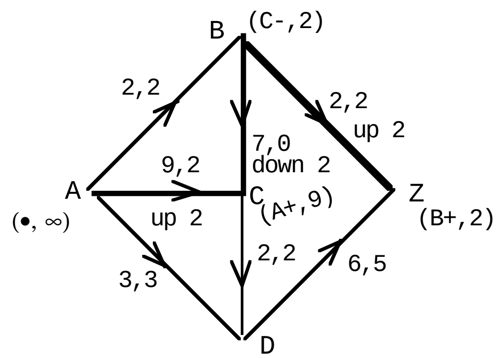
Augment along that path



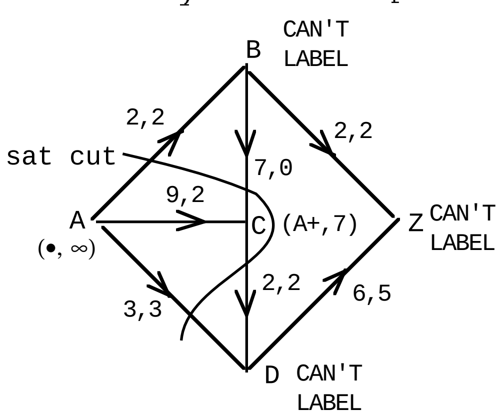
round 2 Label a path to Z



Augment along that path



round 3 Try to label a path to Z

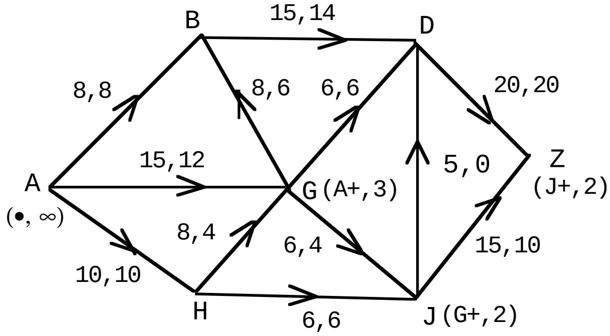


The cut separating the labeled vertices A, C from the can't-be-labeled B, D, Z is a saturated cut.

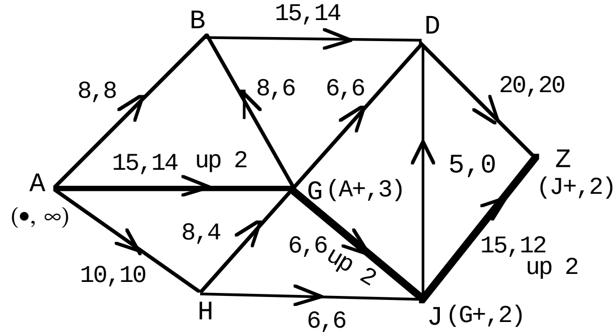
Max flow = flow on sat cut = 2 + 2 + 3 = 7

(d) *round 1*

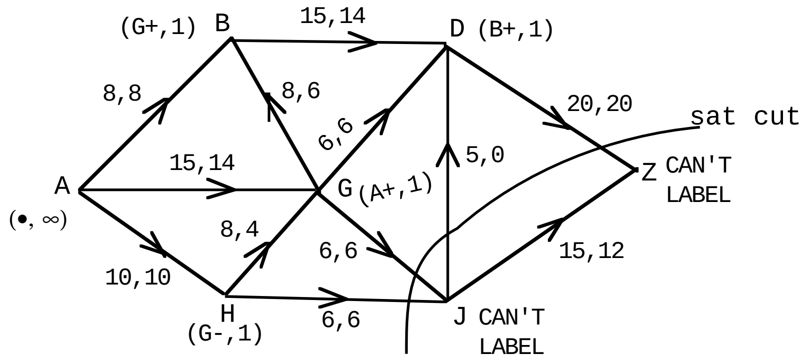
Label a path to Z



Augment along that path



round 2 Try to label a path to Z



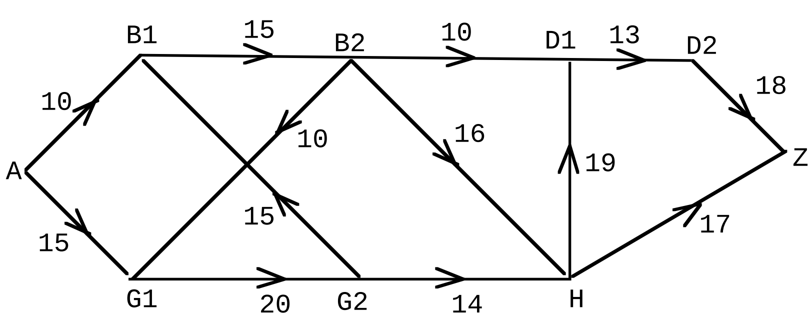
The cut separating the labeled vertices A, B, D, G, H from the can't-be-labeled J, Z is a saturated cut.

Max flow = flow on sat cut = 20 + 6 + 6 = 32 (also equals out of A, also equals into Z).

9. Replace vertex B by B_1 (receiving department for B) and B_2 (shipping department for B) and gave edge B_1B_2 edge weight 15. Similarly for D and G.

Furthermore for each edge *entering* B there is a corresponding edge entering B_1; for each edge *leaving* B there is a corresponding edge leaving B_2.

For example, corresponding to edge G-to-B with capacity 15 in the original there is edge G_2-to-B_1 with capacity 15 in the new version. Corresponding to edge B-to-G with capacity 10 in the old there is edge B_2-to-G_1 with capacity 10 in the new.



SOLUTIONS review problems for Chapter 3

1.(a) I'll use Prim's algorithm. You can start anywhere but I'll start from vertex B.

	E	F	G	I	S	T
status	x	x	x	x	x	x
link	B	B	B	B	B	B
cost	119	174	198	51	198	58
status	x	x	x	√	x	x
link	B	I	I	B	I	B
cost	119	121	153	51	140	58
status	x	x	x	√	x	√
link	T	I	I	B	I	B
cost	113	121	153	51	140	58
status	√	x	x	√	x	√
link	T	I	I	B	I	B
cost	113	121	153	51	140	58
status	√	√	x	√	x	√
link	T	I	F	B	F	B
cost	113	121	132	51	79	58
status	√	√	x	√	√	√
link	T	I	S	B	F	B
cost	113	121	58	51	79	58

Pick G and you're finished. My highway system consists of edges ET, FI, GS, IB, SF, TB. Its length is 113 + 121 + 58 + 51 + 79 + 58 = 480

(b) Begin with I, E and edge IE to satisfy the requirement and then continue being greedy as in Prim's algorithm

	B	E	F	G	I	S	T
status	x	√	x	x	√	x	x
link	I	I	I	I		I	I
cost	51	168	121	153		140	71
status	√	√	x	x	√	x	x
link	I	I	I	I		I	B
cost	51	168	121	153		140	58
status	√	√	x	x	√	x	√
link	I	I	I	I		I	B
cost	51	168	121	153		140	58
status	√	√	√	x	√	x	√
link	I	I	I	F		F	B
cost	51	168	121	132		79	58
status	√	√	√	x	√	√	√
link	I	I	I	S		F	B
cost	51	168	121	58		79	58

Pick G and you're finished. Use edges BI, EI, FI, GS, SF, TB.

2. The shortest distance from A to E *using B, C, F (the picked vertices) as the only possible intermediates* is 6.

(Since E is not picked yet, you might get a shorter path from A to E on a later round, as more vertices get picked and are available to be intermediates.)

3. (a) (i) There is a path from v_3 to v_5 using only v_1, v_2, v_3, v_4 as possible intermediates.

(ii) No path from v_3 to v_5 using only v_1, v_2, v_3, v_4 as possible intermediates.

(b) There is a path from v_2 to v_3 using only $v_1, \ldots, v_6$ as possible intermediates. So (i) and (ii) are both true.

But (iii) is not necessarily true: There is a path using only $v_1, \ldots, v_6$ as *possible* intermediates. The intermediates are not *required.* There could be a path from v_2 to v_3 that only goes through v_5; or that goes directly with no intermediates.

(c) (i) Not correct. There is no path using only $v_1, \ldots, v_6$ as possible intermediates. But there may be a path if other intermediates are allowed; i.e., a 1 may turn up in a later round.

(ii) True. There's no path using just $v_1, \ldots, v_6$ as possible intermediates. So if there's a path at all you have to step on at least one of v_7–v_{10}.

4.(a) Here's my algorithm.

See if the entries in row i are 1's, except for the diagonal entry which is irrelevant.
Do the same for col i.
If one of them isn't 1 then v_i is not a hub.
If all of them are 1's, then v_i is a hub.
Do this for i = 1, ..., n.

(b) Use the comparison "is it a 1" as the unit of measure.
For each i, it takes, at worst, 2n comps (since there are 2n entries in the i-th row and i-th col).

Since the algorithm ignores the diagonal entries then there are actually only n−1 entries to look at in each row and column so that worst there are 2(n−1) comps but it didn't hurt to use 2n.
To do this n times takes at worst $2n^2$ comparisons. More precisely, the number of comps is $2n(n-1)$, i.e., $2n^2-2n$. (But $2n^2$ and $2n^2 - 2n$ have the same order of magnitude so it doesn't matter which you use.)
The running time is $O(n^2)$. Not plain n^2 because this was an "at worst" and not $O(2n^2)$ because you always give the simpler answer n^2.

5. $M_1 = M$ In M, I B-added row 1 to rows 3 and 4 but nothing changed

$$M_2 = \begin{matrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{matrix}$$ In M_1, add row 2 to the other rows

$$M_3 = \begin{matrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{matrix}$$ Add row 3 to all the other rows

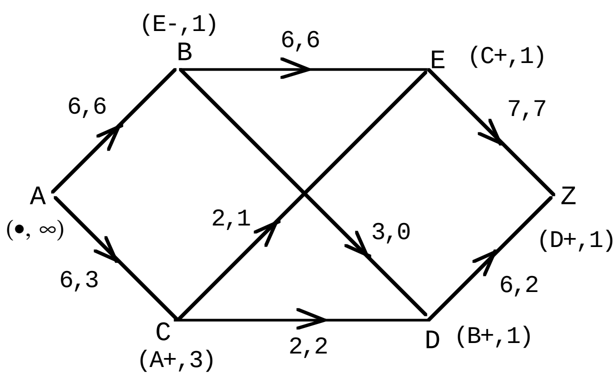
$M_\infty = M_4 = M_3$

6. Pick F and update.
dist F + FA = 9, dist A = 10, change
dist F + FD = 15, dist D = ∞, change
No changes otherwise.

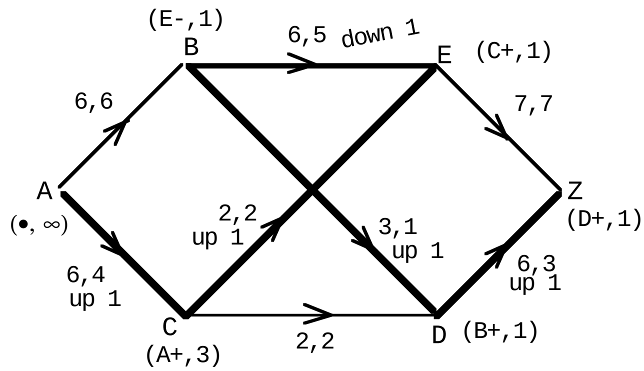
A	B	C	D	E	F	G
x	x	x	x	√	√	x
F	P	P	F	P	E	P
9	∞	∞	15	4	7	∞

Min dist from P to E is 4. The minimal-weight path is PE.
Min dist from P to F is 7. The minimal-weight path is PEF.
Min dist from P to A using E and F as the only possible intermediates is 9 (the path itself is AFEP). But this is not necessarily *the* min distance (allowing any vertex as an intermediate). Actually it will turn out to be the min distance because A is about to be picked on the next round but until you look ahead you don't know that.
Min dist from P to D using E and F as the only possible intermediates is 15 (pat itself is DF) but *the* min distance (not caring who is intermediate) might turn out to be less.

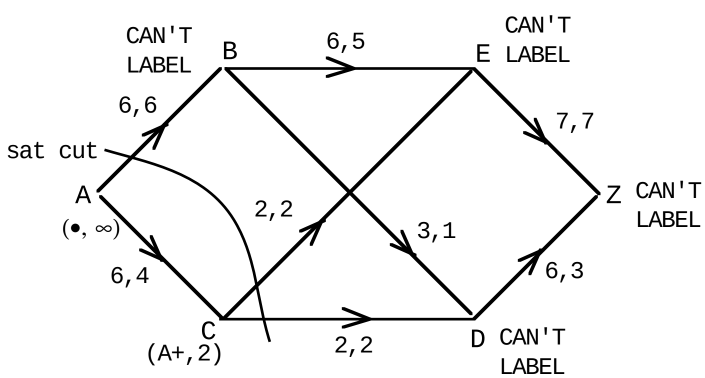
7. *round 1* Label a path to Z



Augment along that path



round 2 Try to label a path to Z



max flow = 6 + 2 + 2 = 10

8. Here's the overall plan.
Get a new matrix by changing all non-zero entries to 1's (Fig A).

$$\begin{matrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{matrix}$$

FIG A

Then use the matching algorithm to get a max set of marriages.
If you get 5 marriages then rearrange the rows as follows:

new row 1 Choose the row of the girl who is married to B1.
new row 2 Choose the row of the girl who is married to B2
...
new row 5 Choose the row of the girl who is married to B5

In the new matrix, this puts 1's (actually boxed 1's, representing marriages) on the diagonal.

Rearrange the rows correspondingly in the old matrix and you will have non-0's on the diagonal.

If you don't get 5 marriages from Fig A then no rearrangement in the original matrix can keep the diagonal zero-free.

Here's what I got when I ran the matching algorithm, breadth first, starting with the marriages in Fig A.

The first two rounds marry up G_1B_1, G_2B_2.

round 3

```
G3-----B2—— G2

G4- - - B1—— G1
   - - - B5  FREE
```

So far we have G_1B_1, G_2B_2, G_4B_5.

round 4

```
G3-----B2—— G2

G5- - - B1—— G1
   - - - B3  FREE
```

So far we have G_1B_1, G_2B_2, G_4B_5, G_5B_3.

round 5

```
G3---B2—— G2---B3—— G5- - - B1—— G1
                       - - - B4  FREE
```

Final matching is G_1B_1, G_2B_3, G_3B_2, G_4B_5, G_5B_4

We got a complete matching so the original matrix can be successfully rearranged. Like this:

G1, G3, G2, G5, G4 are married to B1, B2, B3, B4, B5 respectively.

So line up the rows of the original matrix in the order row 1, row 3, row 2, row 5, row 4 to get the following rearrangement with non-zeros on the main diagonal.

$$\begin{matrix} 6 & 0 & 7 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 4 & 3 & 0 & 0 \\ 1 & 1 & 2 & 3 & 4 \\ 1 & 0 & 0 & 0 & 8 \end{matrix}$$

SOLUTIONS Section 4.1

1. (a) Look at strings of 1's and 2's.

Let y_n be the number of strings whose sum is n (call them good n–strings)

Each good n–string begins with 1 or 2.

So each good n–string (where $n \geq 3$) is one of the following two types.

(i) 1 followed by a string with sum n–1, i.e., a good (n–1)-string

(ii) 2 followed by a string with sum n–2, i.e., a good (n–2)-string

There are y_{n-1} strings of type (i).

There are y_{n-2} strings of type (ii).

So

$$y_n = y_{n-1} + y_{n-2} \text{ for } n \geq 3$$

This is a second order rr so I need two IC:

$y_1 = 1$ (only the string 1 is OK)

$y_2 = 2$ (the strings 2, 11 are OK)

(b) $y_3 = y_2 + y_1 = 2 + 1 = 3$

$y_4 = y_3 + y_2 = 3 + 2 = 5$

$y_5 = y_4 + y_3 = 5 + 3 = 8$

$y_6 = y_5 + y_4 = 8 + 5 = 13$

There are 13 ways to climb a staircase with 6 steps.
(Check: The thirteen strings are 1221, 2112, 2211, 1122, 1212, 2121, 11112, 11121, 11211, 12111, 21111, 111111, 222)

2. Let u_n be the number of n-words with an even number of A's (good n-words).

warning Don't leave this line out. It doesn't make sense to have a recursion relation involving u_n unless you state at the beginning what u_n represents.

The good n-words (with $n \geq 2$) split into these types:

(a) $\overline{A}$ followed by a good (n–1)-word
(b) A followed by a *bad* (n–1)-word, i.e., an (n–1)-word with an *odd* number of A's

There are $25u_{n-1}$ words of type (a) (25 possibilities for the non-A and u_{n-1} possibilities for the good (n–1)-word).

The number of words of type (b)
= total number of (n–1)-words – number of good (n–1)-words
= $26^{n-1} - u_{n-1}$

So $u_n = 25u_{n-1} + 26^{n-1} - u_{n-1}$, i.e.,

$u_n = 26^{n-1} + 24u_{n-1}$ for $n \geq 2$.

This is first order rr. Need one IC.
The IC is $u_1 = 25$ (each of the 1-words B,C,...,Z has an even number of A's).

3.(a) There are $10 \cdot u_{n-1}$ strings of type (a) (D can be picked in any of 10 ways and the legal (n–2)-string in any of u_{n-1} ways).

There are $10 \cdot 3 \cdot u_{n-2}$ strings of type (b).

So all in all,

$$u_n = 10u_{n-1} + 30u_{n-2} \text{ for } n \geq 3.$$

This is a second order rr so I need two IC.

$u_1 = 10$ (a string with 1 symbol can be any of the 10 digits)

$u_2 = 10 \cdot 10 = 100$ (a string with 2 symbols must consist of two digits)

(b) $u_3 = 10u_2 + 30u_1 = 1300$

$u_4 = 10u_3 + 30u_2 = 16{,}000$

4. Let a_n be the number of messages of length n.

A message of length n begins with either a or ab or bc.
So a message of length n (where $n \ge 3$) must be one of the following types:

(i) a followed by a message of length n−1
(ii) ab followed by a message of length n−2
(iii) bc followed by a message of length n−2

So

$$a_n = a_{n-1} + 2a_{n-2} \text{ for } n \ge 3$$

This is a second order rr. The two IC are

$a_1 = 1$ (the only message of length 1 is a)

$a_2 = 3$ (the messages of length 2 are aa, ab, bc)

(b)

$a_3 = 3 + 2 \cdot 1 = 5$

$a_4 = 5 + 2 \cdot 3 = 11$

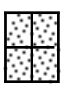
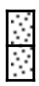
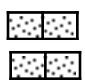
$a_5 = 11 + 2 \cdot 5 = 21$

$a_6 = 21 + 2 \cdot 11 = 43$

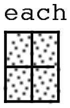
$a_7 = 85, \quad a_8 = 171, \quad a_9 = 341$

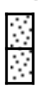
There are 341 messages with 9 letters in them.

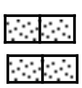
5. The tiling of an n-board must begin, say at the left end of the board, with

 or  or 

So for $n \ge 3$, each tiling must be one of the following three types:

(i)  plus a tiling for an (n−2)-board

(ii)  plus a tiling for an (n−1)-board

(iii)  plus a tiling for an (n−2)-board

The number of tilings of type (i) is y_{n-2}

The number of tilings of type (ii) is y_{n-1}

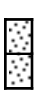
The number of tilings of type (iii) is y_{n-2}

So

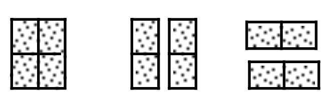
$$y_n = 2y_{n-2} + y_{n-1} \text{ for } n \ge 3$$

This is second order rr. The two IC are

$y_1 = 1$, $y_2 = 3$ (see the diagram below)



one way to tile a 1-board



3 ways to tile a 2-board

6. Let a_n be the number of n-symbol sentences.

For $n \ge 3$, a sentence must begin with two letters or with a letter followed by a blank.

So every sentence of length n, $n \ge 3$, must be one of these two types:

(a) letter followed by # followed by an (n−2)-symbol sentence

(b) letter followed by an (n−1)-symbol sentence

For example, the 5-symbol sentence Q#CAT is Q# followed by the 3-symbol sentence CAT so it's type (a). The 7-symbol sentence THE#CAT is T followed by the 6-symbol sentence HE#CAT so it's type (b).

There are $26a_{n-2}$ sentences of type (a)

There are $26a_{n-1}$ sentences of type (b)

So

$$a_n = 26a_{n-1} + 26a_{n-2} \text{ for } n \ge 3$$

This is second order rr. The two IC are

$a_1 = 26$ (any letter is a 1-symbol sentence)

$a_2 = 26 \cdot 26$ (any string of 2 letters is OK)

7. Each good n-string must be one of these two types:

(a) nonzero followed by a good (n−1)-string

(b) 0 followed by *any* string of 0's, 1's, 2's of length n−1

For example, the good 6-string 020012 is 0 followed by the string 20012 so it's type (b). The good 7–string 3012202 is 3 followed by the good 6–string 012202 so it's type (a).

There are $3a_{n-1}$ strings of type (a) (the nonzero can be picked in 3 ways and the good (n−1)-string in a_{n-1} ways).

There are 3^{n-1} strings of type (b) (each spot in the (n−1)-string can be picked in 3 ways).

So

$$a_n = 3a_{n-1} + 3^{n-1} \text{ for } n \ge 2 \quad \text{(a first order rr)}$$

The IC is

$a_1 = 4$ (there are four good strings of length 1, namely 0,1,2,3)

Then

$$a_2 = 3a_1 + 3^1 = 12 + 3 = 15$$

(check: The 15 strings are 00, 01, 02, 10, 11, 12, 13, 20, 21, 22, 23, 30, 31, 32, 33)

$$a_3 = 3a_2 + 3^2 = 45 + 9 = 54$$

$$a_4 = 3a_3 + 3^3 = 162 + 27 = 189$$

8. (a) The rr is $b_n = 2^{n-2} + 2b_{n-1}$ for $n \ge 3$. But it's wrong because the two categories overlap. The string 00100 is in both categories.

(b) The rr is $b_n = 2^{n-2} + b_{n-1}$ for $n \ge 3$. But it's wrong because the two categories don't include all good (n−1)-strings. The good string 0100 is in neither category.

(c) Every good string must begin with either 1 or 01 or 00.

So every good n–string must be one of these three types

(i) 1 followed by a good (n−1)-string

(ii) 01 followed by a good (n−2)-string

(iii) 00 followed by any (n−2)-string

There are b_{n-1} strings of type (i).

There are b_{n-2} strings of type (ii).

There are 2^{n-2} strings of type (iii) (each of the n−2 spots can be filled in 2 ways).

So

$$b_n = b_{n-1} + b_{n-2} + 2^{n-2} \text{ for } n \ge 3$$

This is a second order rr. The two IC are

$b_1 = 0$ (can't have consecutive 0's if you have only one digit)

$b_2 = 1$ (the only good string is 00)

So

$$b_3 = b_2 + b_1 + 2^1 = 1 + 0 + 2 = 3$$

$$b_4 = b_3 + b_2 + 2^2 = 3 + 1 + 4 = 8$$

$$b_5 = b_4 + b_3 + 2^3 = 8 + 3 + 8 = 19$$

9. (a) There are n spots to fill.
First can be filled in any of 4 ways.
Second can be filled in any of 3 ways (any letter except the one in the 1st spot)
Third can be filled in any of 3 ways (any letter except the one in the 2nd spot)
etc.

Answer is $y_n = 4\cdot 3^{n-1}$

(b) *method 1*

Each good n−string begins with A or B or C or D. So it must be one of the following four types.

(i) A followed by a good (n−1)-string beginning with $\overline{A}$
(ii) B followed by a good (n−1)-string beginning with $\overline{B}$
(iii) C followed by a good (n−1)-string beginning with $\overline{C}$
(iv) D followed by a good (n−1)-string beginning with $\overline{D}$

By symmetry, of all the good (n−1)-strings, $\frac{1}{4}$ begin with A, $\frac{1}{4}$ begin with B, $\frac{1}{4}$ begin with C and $\frac{1}{4}$ begin with D.

So the number of strings of type (i) is $\frac{3}{4}\, y_{n-1}$ and similarly for (ii)−(iv).

So

$$y_n = 4\cdot \frac{3}{4}\, y_{n-1}$$
$$y_n = 3y_{n-1} \text{ for } n \ge 2$$

This is a first order rr. The IC is

$y_1 = 4$ (the good strings A,B,C,D)

method 2

Every good n-string begins with a good (n−1) string followed by one of the three letters that the good (n−1) string didn't end with.

That makes $y_n = 3y_{n-1}$

(c) Part (a) says

$$y_n = 4\cdot 3^{n-1}$$

$$3y_{n-1} = 3\cdot 4\cdot 3^{n-2} = 4\cdot 3^{n-1}$$

So y_n does equal $3y_{n-1}$. And $y_1 = 4\cdot 3^0 = 4$. So the formula in (a) satisfies the rr and IC from (b).

10. A good subset of S_n either contains the last number n or it doesn't contain n. (The same argument works if you focus on any one particular number. It doesn't have to be the last one, n).

So a good subset must be one of the following types:

(i) It contains n in which case it can't contain n−1 so the rest of it is a good subset of $\{1,..., n-2\}$.

(ii) It doesn't contain n in which case it must be a good subset of $\{1, ..., n-1\}$.

For example, look at $S_7 = \{1, 2, ..., 7\}$.

Some good subsets of type (i) are $\{1, 7\}$, $\{7\}$, $\{2, 4, 7\}$.

Some good subsets of type (ii) are $\{2, 5\}$ and $\{1, 4, 6\}$.

There are y_{n-2} subsets of type (i).

There are y_{n-1} subsets of type (ii).

So $y_n = y_{n-1} + y_{n-2}$ for $n \geq 3$.

This is a second order rr so I need two IC>

$S_1 = \{1\}$. It has good subsets ϕ and $\{1\}$.

$S_2 = \{1, 2\}$. It has good subsets ϕ, $\{1\}$, $\{2\}$.

So the IC are $y_1 = 2$, $y_2 = 3$

Then $y_3 = 5$, $y_4 = 8$, $y_5 = 13$, $y_6 = 21$, $y_7 = 34$, $y_8 = 55$.

11. money after n months = old principal after n−1 months + new \$200 deposit + interest on old principal

So

$$u_n = u_{n-1} + 200 + .005\, u_{n-1}$$

$$u_n = 1.005\, u_{n-1} + 200 \text{ for } n \geq 1 \text{ (first order rr) with IC } u_0 = 1000$$

12.(a) See Section 1.10.

First do a new problem where the pairs have names, say Pair 1, Pair 2, ..., Pair n. Then the pairs can be used as the slots and the number of ways to divide the 2n people into these pairs is $\binom{2n}{2}\binom{2n-2}{2}\cdots\binom{4}{2}$

In the original problem the pairs did not have names.

So answer to original problem = $\dfrac{\binom{2n}{2}\binom{2n-2}{2}\cdots\binom{4}{2}}{n!}$

Cancels down nicely to $\dfrac{(2n)!}{(2!)^n \; n!}$

(b) It double counts. It counts these two outcomes as different when they are really the same. Say n = 8

outcome 1 Pick P_1P_7 as the initial pair.

Divide the others into P_2P_3, P_4P_5, P_6P_8.

outcome 2 Pick P_2P_3 as the initial pair.

Divide the others into P_1P_7, P_4P_5, P_6P_8.

(c) To pair up the 2n people, first pair John with one of the 2n−1 others. This can be done in 2n−1 ways.

Then divide up the 2n−2 others into pairs. This can be done in a_{n-1} ways.

So

(*) $$a_n = (2n-1)\, a_{n-1} \text{ for } n \geq 2$$

This is a first order rr. The IC is $a_1 = 1$
(Remember that a_1 is the number of ways of pairing up TWO people and it can be done in only one way.)

(d) Let $a_n = \dfrac{(2n)!}{(2!)^n \; n!}$. Then $a_{n-1} = \dfrac{(2[n-1])!}{(2!)^{n-1} \; (n-1)!} = \dfrac{(2n-2)!}{(n-1)! \; 2^{n-1}}$.

Substitute into (*) and see if it works.

$$\text{RHS of } (*) = (2n-1) \frac{(2n-2)!}{(n-1)! \; 2^{n-1}} = \frac{(2n-1)!}{(n-1)! \; 2^{n-1}}$$

$$\text{LHS of } (*) = \frac{(2n)!}{(2!)^n \; n!} = \frac{2n(2n-1)!}{n! \; 2^n} = \frac{(2n-1)!}{(n-1)! \; 2^{n-1}}$$

LHS = RHS so it checks.

13. To get the last switch ON, you must go through these stages.

(a)	OFF OFF ... OFF OFF OFF	(the starting configuration)
(b)	OFF OFF ... OFF ON OFF	(now you're ready to flip the last switch)
(c)	OFF OFF ... OFF ON ON	(You finally flipped the last switch)
(d)	OFF OFF ... OFF OFF ON	(You turned the preceding switch OFF)

To go from (a) to (b) takes u_{n-1} flips.

To go from (b) to (c) takes 1 flip.
Going from (c) to (d) takes the same number of flips as going from (a) to (b), namely u_{n-1}.
So $u_n = 2u_{n-1} + 1$ for $n \geq 2$ (first order rr) with IC $u_1 = 1$

SOLUTIONS Section 4.2

1. (a) $\lambda^2 - 3\lambda - 10 = 0$, $\lambda = -2, 5$, $y_n = A(-2)^n + B\,5^n$

(b) $\lambda^2 + 3\lambda - 4 = 0$, $\lambda = 1, -4$, $y_n = A\cdot 1^n + B(-4)^n = A + B(-4)^n$

(c) $2\lambda^2 + 2\lambda - 1 = 0$, $\lambda = \dfrac{-1 \pm \sqrt{3}}{2}$

$$y_n = A\left[\frac{-1+\sqrt{3}}{2}\right]^n + B\left[\frac{-1-\sqrt{3}}{2}\right]^n$$

(d) this is the same equation as (b) so it has the same answer

2. $\lambda^2 + 2\lambda - 15 = 0$, $\lambda = -5, 3$, gen $y_n = A(-5)^n + B\,3^n$

Need $A + B = 0$, $-5A + 3B = 1$. So $A = -\frac{1}{8}$, $B = \frac{1}{8}$. Answer is $y_n = -\frac{1}{8}(-5)^n + \frac{1}{8}\cdot 3^n$

3. (a) $y_{n+2} = y_{n+1} + 6y_n$ so

$y_2 = y_1 + 6y_0 = 0 + 6\cdot 1 = 6$

$y_3 = y_2 + 6y_1 = 6 + 6\cdot 0 = 6$

(b) $\lambda^2 - \lambda - 6 = 0$, $\lambda = 3, -2$, gen $y_n = A\,3^n + B(-2)^n$

Need $1 = A + B$, $0 = 3A - 2B$ so $A = \frac{2}{5}$, $B = \frac{3}{5}$. Sol is $y_n = \frac{2}{5}\cdot 3^n + \frac{3}{5}(-2)^n$

(c) $y_3 = \frac{2}{5}\cdot 3^3 + \frac{3}{5}(-2)^3 = 6$

4. $\lambda = \dfrac{1 \pm \sqrt{5}}{2}$, gen $y_n = A\left(\dfrac{1+\sqrt{5}}{2}\right)^n + B\left(\dfrac{1-\sqrt{5}}{2}\right)^n$

To get the IC you need

$$A + B = 0,$$
$$\frac{1}{2}A(1+\sqrt{5}) + \frac{1}{2}B(1-\sqrt{5}) = 1,$$

The solution is $A = \dfrac{1}{\sqrt{5}}$, $B = -\dfrac{1}{\sqrt{5}}$ so the final answer is

$$y_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$$

In case you don't believe that this formula actually produces *integers* here is a table of the first 9 values.

```
Fib[n_]:=1/Sqrt[5]((1 + Sqrt[5])/2)^n -1/Sqrt[5]  ((1 - Sqrt[5])/2)^n
Table[Fib[n]//Simplify,{n,0,9}]
{0, 1, 1, 2, 3, 5, 8, 13, 21, 34}
```

5. (a) Sequence is 5, 7, 9, 11, 13, ... Pattern looks like $y_n = 2n + 3$

(b) $y_{n+1} = \dfrac{y_{n+2} + y_n}{2}$, $y_{n+2} - 2y_{n+1} + y_n = 0$, $\lambda = 1, 1$, gen $y_n = A + Bn$.

To get the IC $y_1 = 5$, $y_2 = 7$ need $A + B = 5$, $A + 2B = 7$

So $A = 3$, $B = 2$. Answer is $y_n = 3 + 2n$

6. Substitute $n3^n$ into the rr.

$$\begin{aligned} \text{LHS} &= (n+2)3^{n+2} - 6(n+1)3^{n+1} + 9n3^n \\ &= (n+2)3^2\cdot 3^n - 6(n+1)3\cdot 3^n + 9n3^n \\ &= (9n + 18 - 18n - 18 + 9n)3^n = 0 \quad \text{So it checks.} \end{aligned}$$

7. rr is $y_{n+2} - 4y_{n+1} + 4y_n = 0$

$\lambda^2 - 4\lambda + 4 = 0$, $\lambda = 2,2$, gen $y_n = A2^n + Bn2^n$

To get $y_0 = 0$ need $0 = A$

To get $y_1 = 2$ need $2 = 2A + 2B$, $B = 1$

Answer is $y_n = n2^n$

8. The equ is really $y_{n+2} - 2y_n = 0$ so $\lambda^2 + 2 = 0$, $\lambda = \pm\sqrt{2}$

gen $y_n = A\ (\sqrt{2})^n + B(-\sqrt{2})^n$

9. (a) $y_n = A(-3)^n + B\,4^n + Cn4^n$

(b) $y_n = A\ 5^n + Bn\ 5^n + Cn^2\ 5^n + Dn^3\ 5^n + E\,2^n$

(c) $y_n = A + Bn + Cn^2 + D\,6^n + E(-1)^n$

10. $\lambda = 1,1,2$, $(\lambda-1)^2(\lambda-2) = 0$, $\lambda^3 - 4\lambda^2 + 5\lambda - 2 = 0$
rr is $y_{n+3} - 4y_{n+2} + 5y_{n+1} - 2y_n = 0$

11. $(\lambda - 1)^3 = 0$, $\lambda = 1,1,1$, gen $y_n = A + Bn + Cn^2$

Need

$A + B + C = 0$
$A + 2B + 4C = 1$
$A + 3B + 9C = 0$

Solution is $A = -3$, $B = 4$, $C = -1$

So $y_n = -3 + 4n - n^2$

12. (a) *by inspection* If $y_1 = 4$ and $y_{n+1} = y_n$ then $y_2 = 4$, $y_3 = 4$ and the solution is $y_n = 4$ for every n

overkill $\lambda = 1$, gen $y_n = A$. Plug in the IC $y_1 = 4$ to get $A = 4$. Sol is $y_n = 4$

(b) *by inspection* If $y_0 = 0$, $y_1 = 0$ and thereafter $y_{n+2} = -\frac{b}{a}y_{n+1} - \frac{c}{a}y_n$ then every term is 0, i.e., $y_n = 0$ for all n

overkill Say the roots of the characteristic equ are λ_1 and λ_2. Then the general sol is $A\,\lambda_1{}^n + B\,\lambda_2{}^n$ (assuming $\lambda_1 \neq \lambda_2$ which is another story). Plug in the IC and you get $0 = A + B$, $0 = A\,\lambda_1 + B\,\lambda_2$,

The only solution for A and B is $A = 0$, $B = 0$ which makes $y_n = 0$.

13. (a) $\lambda = 3,2$, gen $y_n = A3^n + B2^n$

(b) I'm going to compute $5y_{n-1} - 6y_{n-2}$ and see that it comes out to be y_n.

$$5y_{n-1} - 6y_{n-2} = 5A3^{n-1} + 5B2^{n-1} - 6A3^{n-2} - 6B2^{n-2}$$

$$= \frac{5A}{3}\,3^n + \frac{5B}{2}\,2^n - \frac{6A}{3^2}\,3^n - \frac{6B}{2^2}\,2^n \qquad \text{(algebra)}$$

$$= \left(\frac{5A}{3} - \frac{6A}{3^2}\right)3^n + \left(\frac{5B}{2} - \frac{6B}{2^2}\right)2^n \qquad \text{(more algebra)}$$

$$= A3^n + B2^n \qquad \text{(arithmetic)}$$

$$= y_n \quad \text{QED}$$

14. You would need

$$7 = A + B$$
$$5 = 2A + 2B$$

i.e.,

$$A + B = 7$$
$$A + B = 5/2$$

A+B can't be 7 *and* 5/2 so there is no solution. The faux general solution $A2^n + B2^n$ can't be made to satisfy the IC.

$A2^n + B2^n$ is $(A+B)2^n$. There really is only one constant, namely A+B, so it isn't general. And it wasn't enough to satisfy *two* IC.

SOLUTIONS Section 4.3

1. (a) $\lambda = 2, -1$, $h_n = A\,2^n + B(-1)^n$. Try $p_n = C$.

Then $p_{n+1} = C$, $p_{n+2} = C$. Substitute into the rr to get $C - C - 2C = 1$, $C = -\frac{1}{2}$

gen $y_n = A\,2^n + B(-1)^n - \frac{1}{2}$

Plug in the IC: $1 = 2A - B - \frac{1}{2}$, $\quad 3 = 4A + B - \frac{1}{2}$

So $A = \frac{5}{6}$, $B = \frac{1}{6}$. Answer is $y_n = \frac{5}{6}\cdot 2^n + \frac{1}{6}(-1)^n - \frac{1}{2}$

(b) $\lambda = -5, 3$, $h_n = A\,3^n + B(-5)^n$. Try $p_n = Cn + D$

Need $C(n+2) + D + 2\Big[C(n+1) + D\Big] - 15(Cn + D) = 6n + 10$

Equate n coeffs $\quad -12C = 6$, $C = -\frac{1}{2}$

Equate constant terms $\quad 4C - 12D = 10$, $D = -1$

$p_n = -\frac{1}{2}n - 1$

gen $y_n = A\,3^n + B(-5)^n - \frac{1}{2}n - 1$

The IC make $A = \frac{11}{8}$, $B = \frac{5}{8}$, Answer is $y_n = \frac{11}{8}\cdot 3^n + \frac{5}{8}(-5)^n - \frac{1}{2}n - 1$

2. (a) $\lambda^2 - 3\lambda + 1 = 0$, $\lambda = \frac{3 \pm\sqrt{5}}{2}$, $h_n = A\left(\frac{3+\sqrt{5}}{2}\right)^n + B\left(\frac{3-\sqrt{5}}{2}\right)^n$

Try $p_n = C\,4^n$. Need

$$C\,4^{n+2} - 3C\,4^{n+1} + C\,4^n = 10\cdot 4^n$$
$$16C\,4^n - 12C\,4^n + C\,4^n = 10\cdot 4^n$$

Equate coeffs of 4^n: $5C = 10$, $C = 2$

$p_n = 2\cdot 4^n$, gen $y_n = A\left(\frac{3+\sqrt{5}}{2}\right)^n + B\left(\frac{3-\sqrt{5}}{2}\right)^n + 2\cdot 4^n$

(b) The new version of the rr is $y_n - 3y_{n-1} + y_{n-2} = 10\cdot 4^{n-2}$ **warning** Not $10\cdot 4^n$

Same λ equation, same h_n as in part (a).

method 1 for getting p_n Write 4^{n-2} as $4^n\,4^{-2}$ and use $y_n - 3y_{n-1} + y_{n-2} = \frac{10}{16}\cdot 4^n$

Try $p_n = C\,4^n$. Need

$$C\,4^n - 3C4^{n-1} + C4^{n-2} = \frac{10}{16}\,4^n$$
$$C\,4^n - \frac{3}{4}\,C4^n + \frac{3}{16}\,C4^n = \frac{10}{16}\,4^n$$

Match the coeffs of 4^n: $C - \frac{3}{4}C + \frac{3}{16}C = 2$, $\frac{5}{16}C = \frac{10}{16}$, $C = 2$

$p_n = 2\cdot 4^n$, same as in part (a)

Same gen y_n as in part (a).

method 2 for getting p_n (the first method was better so you could skip this)

Leave it $y_n - 3y_{n-1} + y_{n-2} = 10\cdot 4^{n-2}$, try $p_n = C\cdot 4^{n-2}$. Need

$$C\cdot 4^{n-2} - 3C\cdot 4^{n-3} + C\cdot 4^{n-4} = 10\cdot 4^{n-2}$$
$$C\cdot 4^{n-2} - \frac{3}{4}\,C\cdot 4^{n-2} + \frac{1}{16}\,C\cdot 4^{n-2} = 10\cdot 4^{n-2}$$

Match the coeffs of 4^{n-2}: $\frac{5}{16}C = 10$, $C = 32$

$p_n = 32\cdot 4^{n-2} = 32\cdot 4^n\cdot 4^{-2} = 2\cdot 4^n$, same as before.

3. $\lambda = 3, -2$, $h_n = D\,3^n + E(-2)^n$. Try $p_n = An^2 + Bn + C$. Need

$$A(n+2)^2 + B(n+2) + C - \Big(A(n+1)^2 + B(n+1) + C \Big) - 6(An^2 + Bn + C) = 18n^2 + 2$$

Match n^2 coeffs $\quad -6A = 18$
Match n coeffs $\quad 2A - 6B = 0$
Match constant terms $\quad 3A + B - 6C = 2$
So $A = -3$, $B = -1$, $C = -2$

gen $y_n = D\,3^n + E(-2)^n - 3n^2 - n - 2$

The IC make $D = \frac{8}{5}$, $E = -\frac{3}{5}$ Answer is $y_n = \frac{8}{5}\cdot 3^n - \frac{3}{5}(-2)^n - 3n^2 - n - 2$

4. (a) $y_1 = 2$

$y_2 = 2y_1 + 6\cdot 2 = 4 + 12 = 16$

$y_3 = 2y_2 + 6\cdot 3 = 50$

$y_4 = 2y_3 + 6\cdot 4 = 124$

(b) The rr is $y_n - 2y_{n-1} = 6n$. Or you can use $y_{n+1} - 2y_n = 6(n+1)$. As long as you are consistent.

Either way,

$\lambda - 2 = 0$, $\lambda = 2$, $h_n = A\,2^n$.

method 1 for getting p_n Use $y_n - 2y_{n-1} = 6n$

Try $p_n = Bn + C$. Need

$$Bn + C - 2\Big[B(n-1) + C \Big] = 6n$$

Equate n coeffs $\quad -B = 6$, $B = -6$
Equate constant terms $\quad 2B - C = 0$
So $B = -6$, $C = -12$, $p_n = -6n - 12$

Then gen $y_n = A\,2^n - 6n - 12$

The IC makes $2 = 2A - 6 - 12$, $A = 10$. Answer is $y_n = 10\cdot 2^n - 6n - 12$

method 2 for getting p_n Use $y_{n+1} - 2y_n = 6(n+1)$

Try $p_n = Bn + C$. Need

$B(n+1) + C - 2(Bn + C) = 6n + 6$
Equate coeffs of n: $B - 2B = 6$, $B = -6$
Equate constant terms: $B + C - 2C = 6$, $C = -12$
$p_n = -6n - 12$ again

(c) $y_4 = 10\cdot 16 - 6\cdot 4 = 124$

5. (a) $\lambda = 2$, $h_n = A(-2)^n$. Try $p_n = B$ Need

$$B + 2B = 4 \;,\; B = \frac{4}{3}$$

So $p_n = \frac{4}{3}$ and a general solution is $y_n = A(-2)^n + \frac{4}{3}$

(b) Same h_n as part (a).

Try $p_n = C\,4^n$. Need

$$C4^{n+1} + 2C\,4^n = 4^n$$
$$C\cdot 4\cdot 4^n + 2C\,4^n = 4^n$$

The coeff of 4^n on the left is $6C$, on the right it's 1 so set $6C = 1$, $C = \frac{1}{6}$, $p_n = \frac{1}{6}4^n$

The general solution is $y_n = A(-2)^n + \frac{1}{6}\cdot 4^n$

6. (a) Try $p_n = An^4 + Bn^3 + Cn^2 + Dn + E$

(b) Try $p_n = n^4(An^4 + Bn^3 + Cn^2 + Dn + E) = An^8 + Bn^7 + Cn^6 + Dn^5 + En^4$

Step up because A, n, n^2 and n^3 are all homog solutions

(c) h_n Try $p_n = An2^n$ (step up)

(d) Try $p_n = A\ 2^n$

(e) Try $p_n = An^2\ 3^n$ (step up because 3^n and $n3^n$ are both homog sols))

7. $\lambda = \frac{1}{2}$, $h_n = A(\frac{1}{2})^n$. Try $p_n = Bn(\frac{1}{2})^n$ (step up) Need

$$2B(n+1)\ (\tfrac{1}{2})^{n+1} - Bn(\tfrac{1}{2})^n = (\tfrac{1}{2})^n$$

$$2B(n+1)\tfrac{1}{2}\ (\tfrac{1}{2})^n - Bn(\tfrac{1}{2})^n = (\tfrac{1}{2})^n$$

$$B\ (\tfrac{1}{2})^n = (\tfrac{1}{2})^n$$

Equate coeffs of $(\frac{1}{2})^n$. So B = 1, $p_n = (\frac{1}{2})^n$, gen $y_n = A(\frac{1}{2})^n + n\ (\frac{1}{2})^n$

The IC make A = 3. Answer is $y_n = 3(\frac{1}{2})^n + n(\frac{1}{2})^n$

8. $\lambda = 1,1$, $h_n = A + Bn$. Try $p_n = Cn^2$ (step up twice from plain C) Need

$$C(n+2)^2 - 2C(n+1)^2 + Cn^2 = 1$$

The n^2 terms and n terms cancel out.

Equate constant terms: $2C = 1$, $C = \frac{1}{2}$

$p_n = \frac{1}{2}n^2$

gen $y_n = A + Bn + \frac{1}{2}n^2$

The IC make A = 1, $\frac{1}{2} = A + B + \frac{1}{2}$. So B = −1. Answer is $y_n = 1 - n + \frac{1}{2}n^2$

9. $S_{n+1} = S_n + (n+1)^2$, $S_{n+1} - S_n = (n+1)^2$ with IC $S_1 = 1$.

$\lambda = 1$, $h_n = D$. Try $p_n = n(An^2 + Bn + C) = An^3 + Bn^2 + Cn$. (Step up because C is a homog sol.) Need

$$A(n+1)^3 + B(n+1)^2 + C(n+1) - (An^3 + Bn^2 + Cn) = n^2 + 2n + 1$$

The n^3 coeffs drop out.

Equate n^2 coeffs $3A = 1$, $A = \frac{1}{3}$

Equate n coeffs $3A + 2B = 2$, $B = \frac{1}{2}$

Equate constant terms $A + B + C = 1$, $C = \frac{1}{6}$

$p_n = \frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n$

Gen sol is $S_n = D + \frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n$

The IC $S_1 = 1$ makes D = 0.

Answer is $S_n = \frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n$, usually written as $S_n = \frac{n(n+1)(n+2)}{6}$

10.(a) Need $C2^{n+2} - 3C\,2^{n+1} + 2C\,2^n = 6\cdot 2^n$, $4C\,2^n - 6C\,2^n + 2C\,2^n = 6\cdot 2^n$,
The C's cancel out leaving $0 = 6\cdot 2^n$ which can't be satisfied. There is no value of C which makes $C\,2^n$ work.
In general, if you don't step up when you should, some or all of your constants will cancel out and you can't make your trial p_n fit the rr

(b) Need $2A(n+2) + 3A(n+1) + 4An = 18n$, $9An + 7A = 18n$.
Equate n coeffs $9A = 18$, $A = 2$
Equate constant terms $7A = 0$, $A = 0$
Impossible to have A=2 *and* A = 0. So there is no solution of the form An

11. Ugh !!!
A was supposed to be a *constant*. When you substituted p_n into the lefthand side of the rr and got $A(n+2) - 2A(n+1) - 3An$, you treated A as a *constant*. You can't change your mind now and allow A to be $-\frac{5}{4}n$.

To make $-4An$ equal $5n^2$, coeffs of like terms have to match:
The n term on the left has to match the n term on the right.
The n^2 term on the left has to match the n^2 term on the right.
There is no way to make that happen here.

So your conclusion should be that there are no values of the *constant* A that will make $-4An = 5n^2$ so there is no p_n of the form An where A is a constant.

footnote (if you still don't believe me)
$-\frac{5}{4}n^2$ doesn't work in the rr. If you substitute it into the lefthand side you get
$-\frac{5}{4}(n+2)^2 - 2\cdot -\frac{5}{4}(n+1)^2 - 3\cdot -\frac{5}{4}n^2$
which comes out to be $\frac{5}{2}n^2 - \frac{5}{4}$ not $5n^2$.

warning When you determine the constants in p_n, they should come out to be just that, *constants*.
A can't have n's in it.

SOLUTIONS review problems for Chapter 4

1. $\lambda^2 - 9 = 0$, $\lambda = \pm 3$, $h_n = A\,3^n + B(-3)^n$

Try $y_p = Cn^2 + Dn + E$

$C(n+2)^2 + D(n+2) + E - 9(Cn^2 + Dn + E) = 56n^2$

equate n^2 coeffs $-8C = 56$, $C = -7$

equate n coeffs $4C - 8D = 0$, $D = -7/2$

equate constant terms $4C + 2D - 8E = 0$, $E = -35/8$

$p_n = -7n^2 - 7/2\ n - 35/8$

gen $y_n = h_n + p_n = A3^n + B(-3)^n - 7n^2 - 7/2\ n - 35/8$

2. $\lambda = -2$, $h_n = A(-2)^n$. Try $p_n = B\ 7^n$.

Need $2B\ 7^{n+1} + 4B\ 7^n = 6\cdot 7^n$

$14B\ 7^n + 4B\ 7^n = 6\cdot 7^n$

So $18B = 6$, $B = \frac{1}{3}$, $p_n = \frac{1}{3}\cdot 7^n$, gen $y_n = A(-2)^n + \frac{1}{3}\cdot 7^n$

To get the IC we need $5 = -2A + \frac{7}{3}$, $A = -\frac{4}{3}$.

Answer is $y_n = -\frac{4}{3}(-2)^n + \frac{1}{3}\cdot 7^n$

3. $\lambda = \frac{-5 \pm\sqrt{29}}{2}$, $h_n = A\left[\frac{-5+\sqrt{29}}{2}\right]^n + B\left[\frac{-5-\sqrt{29}}{2}\right]^n$

Try $p_n = D$. Need $D + 5D = 6$, $D = \frac{6}{5}$.

$p_n = \frac{6}{5}$

gen $y_n = A\left[\frac{-5+\sqrt{29}}{2}\right]^n + B\left[\frac{-5-\sqrt{29}}{2}\right]^n + \frac{6}{5}$

4. $S_{n+1} = S_n + n+1$, $S_{n+1} - S_n = n+1$ with IC $S_1 = 1$.

$\lambda = 1$, $h_n = A$.

Try $p_n = n(Bn + C)$ (step up) $= Bn^2 + Cn$

Need $B(n+1)^2 + C(n+1) - (Bn^2 + Cn) = n+1$

The n^2 terms drop out on each side

Equate n coeffs $2B + C - C = 1$, $B = \frac{1}{2}$

Equate constant terms $B + C = 1$, $C = \frac{1}{2}$

$p_n = \frac{1}{2}n^2 + \frac{1}{2}n$

gen $S_n = A + \frac{1}{2}n^2 + \frac{1}{2}n$

To get $S_1 = 1$ we need $1 = A + \frac{1}{2} + \frac{1}{2}$, $A = 0$.

Answer is $S_n = \frac{1}{2}n^2 + \frac{1}{2}n$ usually written as $S_n = \frac{n(n+1)}{2}$

5. $\lambda = \pm 3$, $h_n = A\ 3^n + B(-3)^n$. Try $p_n = Dn\ 3^n$ (step up).

Need $D(n+2)\ 3^{n+2} - 9Dn\ 3^n = 5\cdot 3^n$

$9D(n+2)\ 3^n - 9Dn\ 3^n = 5\cdot 3^n$

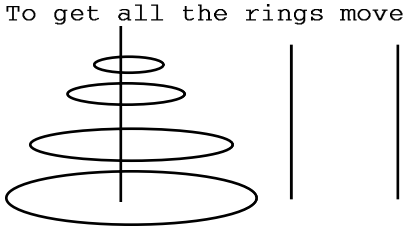
The $n3^n$ terms drop out.

Match the 3^n coeffs: $18D = 5$, $D = \frac{5}{18}$, $p_n = \frac{5}{18}n3^n$

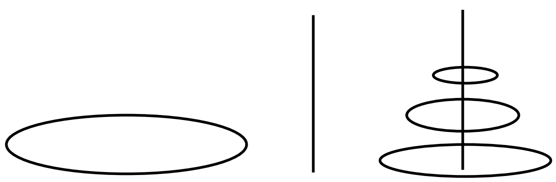
gen $y_n = A\,3^n + B(-3)^n + \frac{5}{18}n3^n$

6. The rr can be written as $y_{n+1} - 2y_n = 0$ and it is only *first* order. It would come with only one IC and its general sol (namely $B\,2^n$) *should* only have one constant. So nothing is wrong.

7.(a) Note that y_n is the minimum number of moves it takes to move n rings from one peg to a second peg, using the third peg for intermediate storage.
To get all the rings moved you have to pass through the following stages

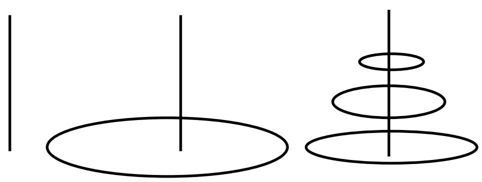


Start here



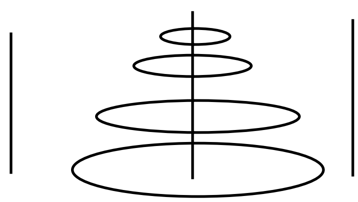
Move the top n-1 rings from peg 1 to peg 3 using peg 2 as storage

Takes y_{n-1} moves



Move the largest ring to peg 2

Takes one move



Move the n−1 rings from peg 3 to peg 2 using peg 1 as storage

Takes y_{n-1} moves

So $y_n = 2y_{n-1} + 1$ for $n \ge 2$. Need one IC.
It only takes one move in a 1-ring game so the IC is $y_1 = 1$.

(b) $\lambda = 2$, $h_n = A\,2^n$. Try $p_n = B$.
Need $B - 2B = 1$, $B = -1$
Gen sol is $y_n = A\,2^n - 1$. To get the IC we need $1 = 2A - 1$, $A = 1$.

Sol is $y_n = 2^n - 1$.

For example to move a 10-ring tower it takes a minimum of $2^{10} - 1$ moves.

8. Every n-tree is one of these five types (see the diagram below).

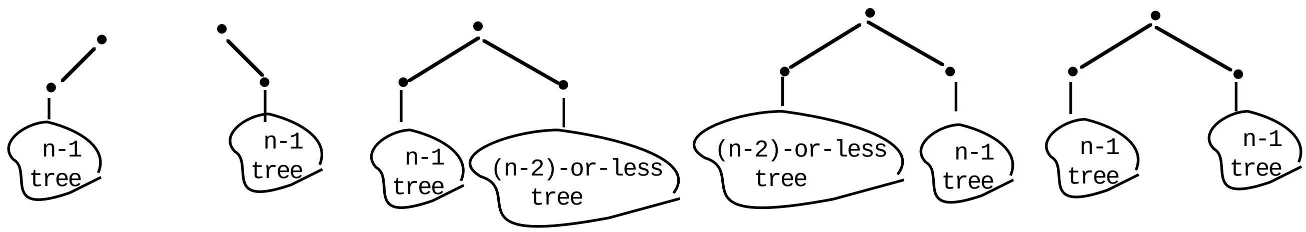
(i) A 1-tree with Left Child followed by an (n−1)-tree

(ii) A 1-tree with a Right Child followed by an (n−1)-tree

(iii) A 1-tree with Both Children, with an (n−1)-tree on the Left Child and either a 0-tree or a 1-tree or ... or an (n−2)-tree on the Right Child

(iv) A 1-tree with Both Children, with an (n−1)-tree on the Right Child and either a 0-tree or a 1-tree or ... or an (n−2)-tree on the Left Child

(v) A 1-tree with both Children, with an (n−1)-tree on each child



(i) (ii) (iii) (iv) (v)

There are y_{n-1} trees of type (i)
There are y_{n-1} trees of type (ii)
There are $y_{n-1} \cdot (y_0 + y_1 + \ldots + y_{n-2})$ trees of type (iii)
(The dangle on the left can be chosen in y_{n-1} ways and the dangle on the right can be chosen in $y_0 + y_1 + \ldots + y_{n-2}$ ways)
There are $y_{n-1} \cdot (y_0 + y_1 + \ldots + y_{n-2})$ trees of type (iv)
There are y_{n-1}^2 trees of type (v)

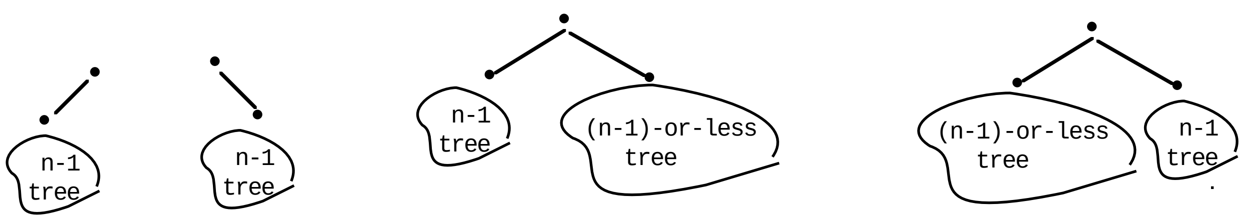
So $y_n = 2y_{n-1} + 2y_{n-1} \cdot (y_0 + \ldots + y_{n-2}) + y_{n-1}^2$
(b) Start with $y_0 = 1$, $y_1 = 3$ Then

$$y_2 = 2y_1 + 2y_1\, y_0 + y_1^2 = 6 + 6 + 9 = 21$$

$$y_3 = 2y_2 + 2y_2 (y_0 + y_1) + y_2^2 = 42 + 42 \cdot 4 + 441 = 651$$

(c) No. It has products $y_0\, y_{n-1}$, $y_1\, y_{n-1}, \ldots, y_{n-2}\, y_{n-1}$ in it so it doesn't have the right pattern for the methods in Sections 2 and 3.

Question Can you say that each n-tree is one of these *four* types.



(i) (ii) (iii)(a) (iv)(a)

Answer
Only if you're careful. These types are exhaustive but not mutually exclusive. Types (iii)(a) and (iv)(a) overlap. In fact the overlap is precisely type (v) above so if you just add the number of trees in (i),(ii),(iii)(a), (iv)(a) you will have counted trees of type (v) twice. So

$$y_n = \text{\# of type (i)} + \text{\# type (ii)} + \text{\# type (iii)(a)} + \text{\# type (iv)(a) MINUS \# type (v)}$$

$$= 2y_{n-1} + 2y_{n-1} \cdot (y_0 + y_1 + \ldots + y_{n-1}) - y_{n-1}^2$$

which agrees algebraically with the other version.

9. Let y_n be the number of ways of filling up n spaces.

Every way must look like one of these

(a) a limo followed by parking in n−2 spaces

(b) a stretch limo followed by parking in n−3 spaces

So $y_n = y_{n-2} + y_{n-3}$ (3rd order rr, need 3 IC)

All in all, $y_1 = 0$, $y_2 = 1$, $y_3 = 1$ and $y_n = y_{n-2} + y_{n-3}$ for $n \geq 4$,

10. Let y_n be the unpaid balance after n months when the interest rate is 1% a month and you make monthly payments of D dollars. Then

unpaid balance this month
= last month's unpaid balance + interest on that unpaid balance − your payment D

So

$$y_{n+1} = y_n + .01\, y_n - D$$

$$y_{n+1} - 1.01 y_n = -D \quad \text{(nonhomog, first order)} \quad \text{with IC} \quad y_0 = 10{,}000$$

$$\lambda - 1.01 = 0,\ \lambda = 1.01,\ h_n = A(1.01)^n$$

Try $p_n = B$ Substitute it into the rr to determine B.

$$B - 1.01B = -D$$
$$B = 100\,D$$

So $p_n = 100\,D$ and gen $y_n = A(1.01)^n + 100\,D$

Plug in the IC $y_0 = 10{,}000$. Get $A = 10{,}000 - 100D$

So $y_n = (10{,}000 - 100\,D)(1.01)^n + 100\,D$

Now arrange to get $y_{36} = 0$:

$$0 = (10000-100D)(1.01)^{36} + 100\,D$$

$$D = \frac{100(1.01)^{36}}{(1.01)^{36} - 1} = \$332.14$$

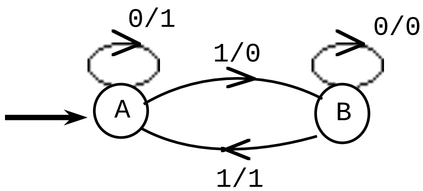
So your monthly payments should be \$332.14

SOLUTIONS Section 5.1

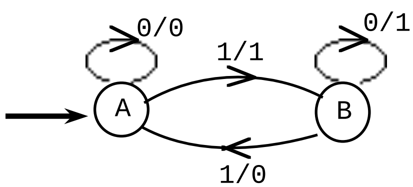
1.

input	0	0	1	1	0	0	0	1	0
output	p	p	q	p	r	q	r	p	p
next state	A	A	B	B	C	B	C	A	A

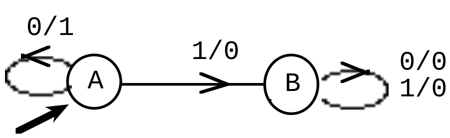
2. (a) State A below remembers that there an even number of 1's so far. The FSM is in state B when there are an odd number of 1's.



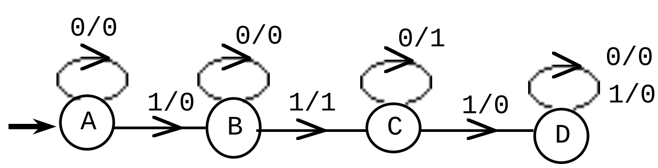
(b)



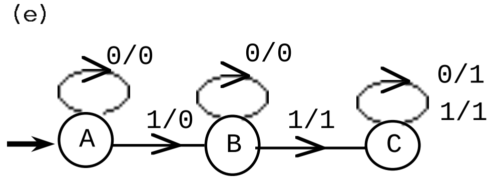
(c)



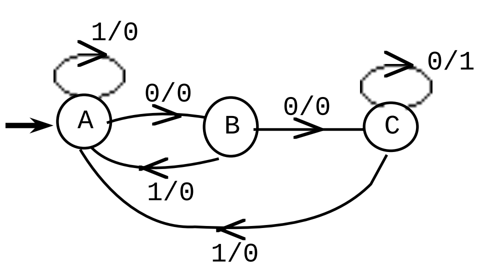
(d)



(e)

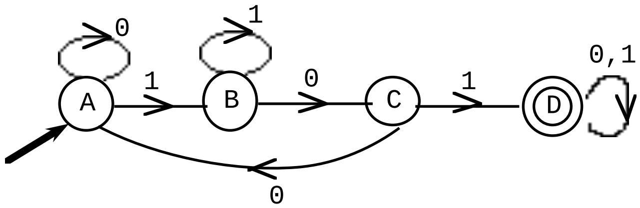


(f)

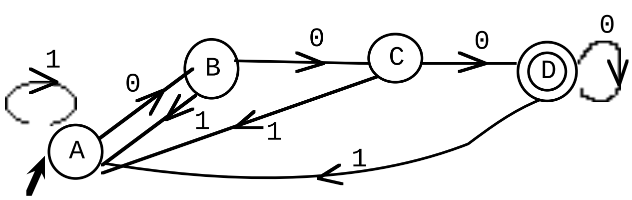


SOLUTIONS Section 5.2

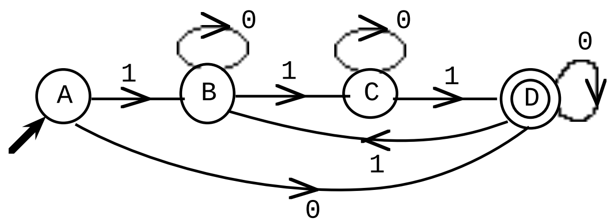
1. (a)



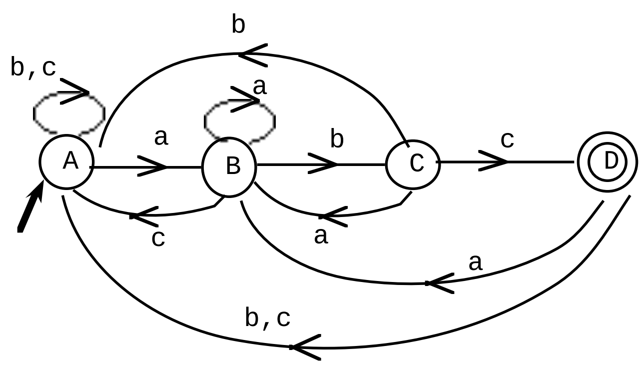
(b)



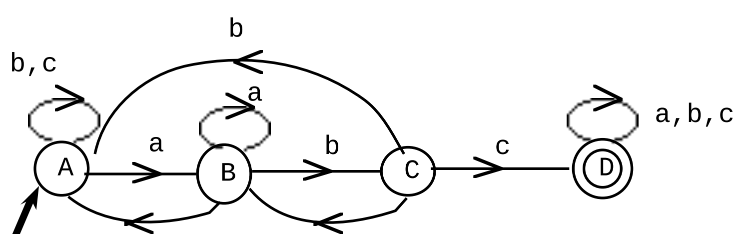
(c)



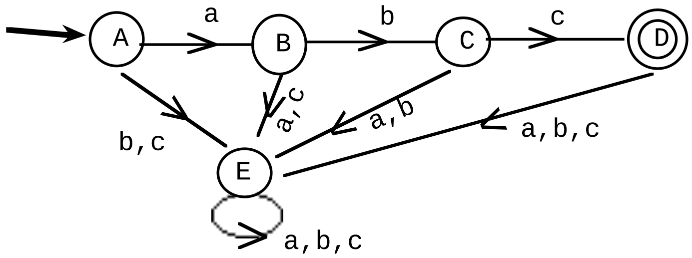
2. (a)



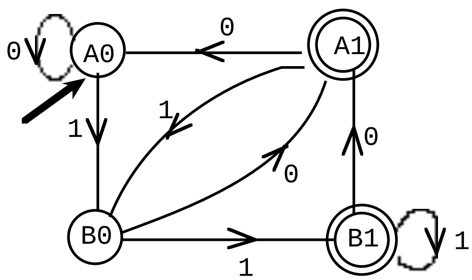
(b) Like the FSM in part (a) but once you are in state D after the first occurrence of acb, the string should stay in D and continue to be accepted no matter what else comes.



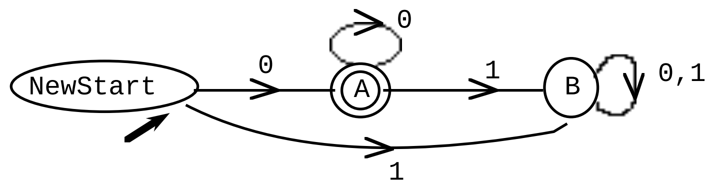
(c)


A
B
C
D
E
a
b
c
b,c
a,c
a,b
a,b,c
a,b,c

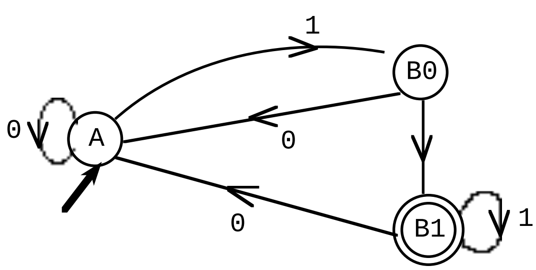
3. (a)


A0
A1
B0
B1
0
0
1
1
0
0
1
1

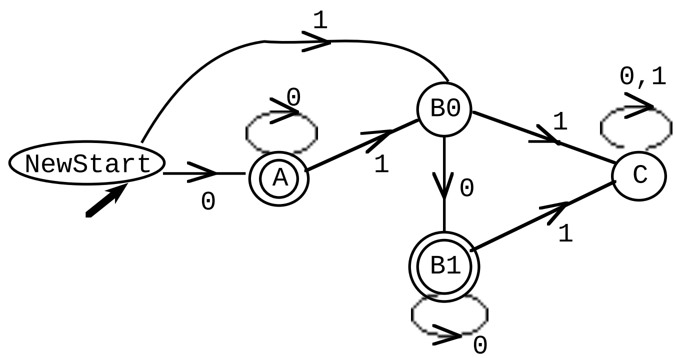
(b)


NewStart
A
B
0
0
1
0,1
1

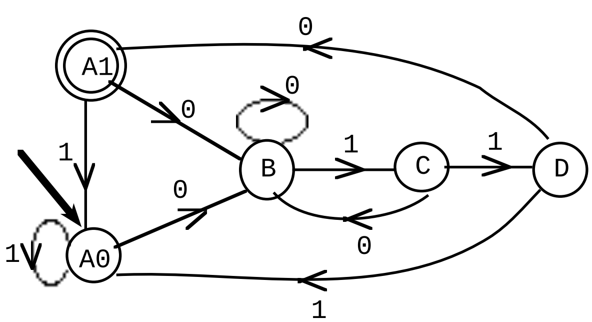
(c)


A
B0
B1
1
0
0
0
1

(d)


NewStart
A
B0
B1
C
1
0
0
1
1
0
0,1
1
0

(e)

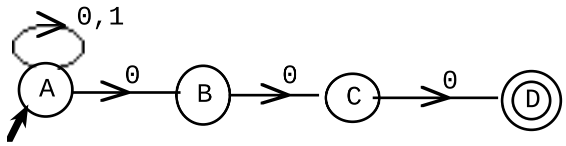

A1
A0
B
C
D
0
0
0
1
1
1
0
0
1
1

SOLUTIONS Section 5.3

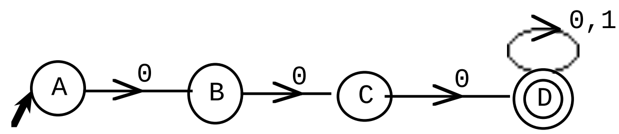
1. (a) Accepted because of the path A A A B
   (b) Not accepted. The only tracks the string could follow are A A C and A B none.

2. Strings ending in 101.

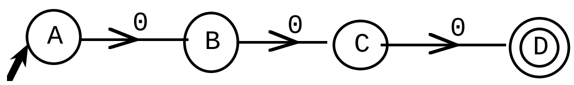
3. (a)



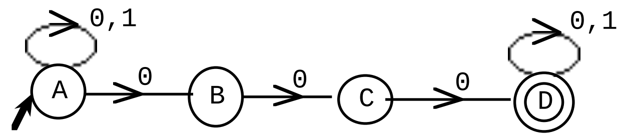
(b)



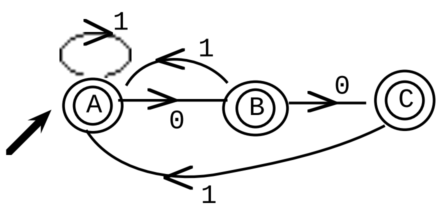
(c)



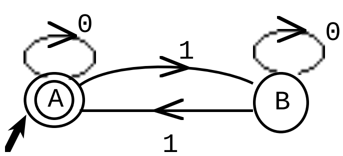
(d)



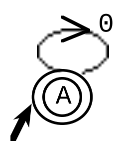
(e)



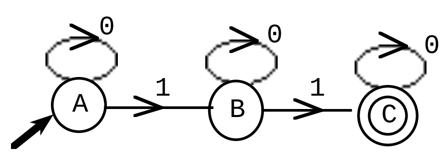
(f)



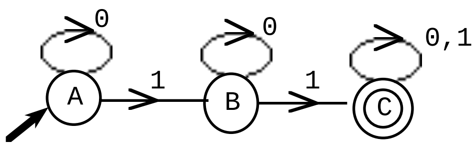
(g)



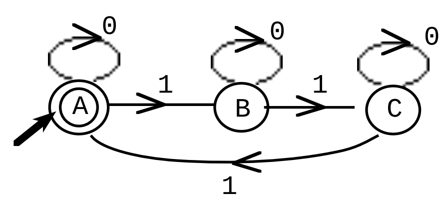
(h)



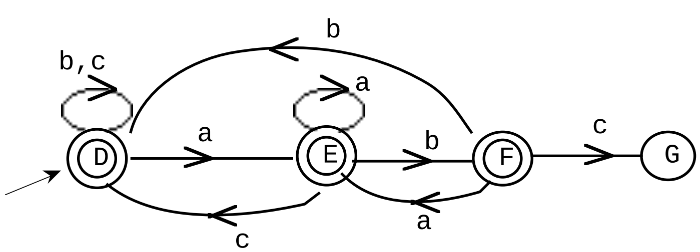
(i)



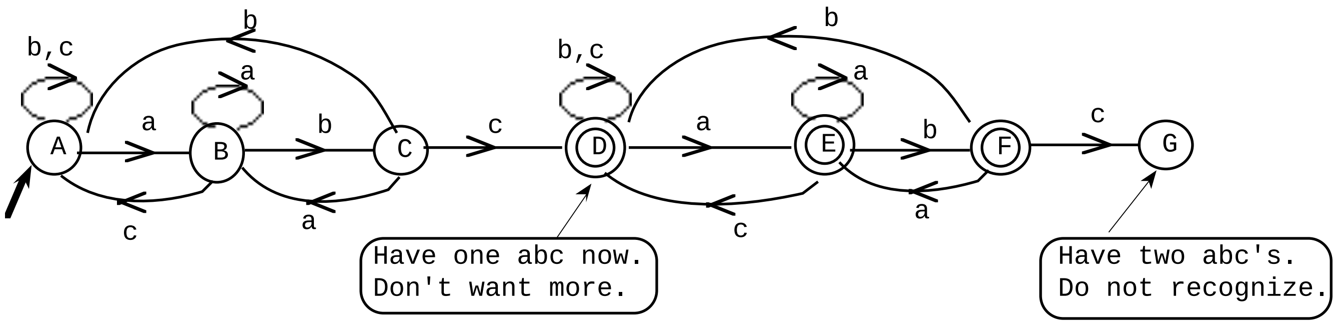
(j)



4. (a)



(b)



(c) Take the answer to (b) and make A, B, C accepting as well as D, E, F. Then the FSM recognizes strings with no abc's and with one abc but not strings with more than one abc.

5. (a) Accepts 011. See path A B C B C
Accepts 0. See path A B C
Doesn't accept 00 (always end up in state D)
Accepts λ since the starting state is accepting.

(b) λ-adjacency matrix

	A	B	C	D
A				
B			1	
C		1		1
D				

λ-reachability matrix

	A	B	C	D
A				
B			1	1
C		1		1
D				

6.

in	0	1	1	1	0
next	C	C,D	C,D	C,D	none

not accepted

7. (a)

in		λ's	1	λ's	1	λ's
next	A	A,B	A,D	A,B,D	A,D	A,B, $\boxed{D}$

accepts 11

in		λ's	0	λ's	0	λ's
next	A	A,B	B,C	B,C,D	B	B

rejects 00

(b)

	A	B	C	D
A	1	1	0	0
B	0	1	0	0
C	0	0	1	1
D	0	1	0	1

(c)

	A	B	C	D
A	1	1	0	0
B	0	1	0	0
C	0	1	1	1
D	0	1	0	1

8. (a) Use Warshall's algorithm on the λ-adjacency matrix. The λ-reachability matrix turns out to be the same as the λ-adjacency matrix

(b)

in		λ's	a	λ's	a	λ's
next	A	A,B,D	C	B,C,D	C	B,C, $\boxed{D}$

accepts aa

9. (a) No λ-moves to worry about. Makes it easier.
The only state λ-reachable from A is A so the starting state in the new DFSM will be named A.

To get the 0-exit from A:
A -> A
Next state is A

To get the 1-exit from A:
A -> A, B
Next state will be named AB.

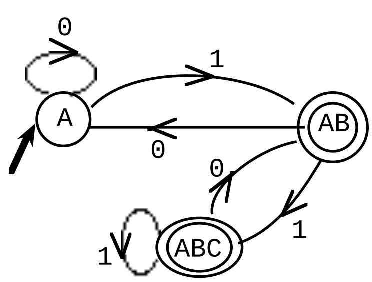
To get the 0-exit from AB:
A -> A
B -> none
Next state is A

To get the 1-exit from AB:
A -> A, B
B -> A, B, C
Next state is ABC

To get the 0-exit from ABC:
A -> A
B -> none
C -> A, B
Next state is AB

To get the 1-exit from ABC:
A -> A, B
B -> A, B, C
C -> who cares
Next state is ABC

Accepting states are those states with a B in the name.



(b) The only state λ-reachable from A is A so the starting state in the new DFSM will be named A.

To get the 0-exit from A:
A -> A, B
Next state is AB

To get the 1-exit from A:
A -> A
Next state is A

To get the 0-exit from AB:
A -> A, B
B -> none
Next state is AB

To get the 1-exit from AB:
A -> A
B -> C -> (λ moves) A, C   WARNING Don't forget the λ moves
Next state is AC

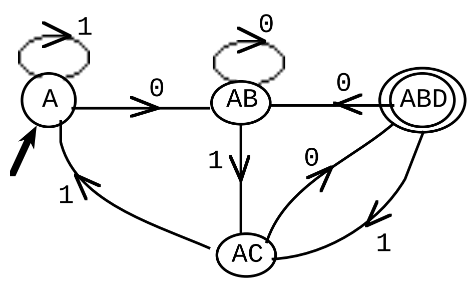
To get the 0-exit from AC:
A -> A, B
C -> D -> (λ moves) B, D
Next state is ABD

To get the 1-exit from AC:
A -> A
C -> none
Next state is A

To get the 0-exit from ABD
A -> A, B
B -> none
C -> none
Next state is AB

To get the 1-exit from ABD
A -> A
B -> C -> (λ moves) A, C
D -> none
Next state is AC

The accepting states are those states with a D in the name.



(c) From A you can λ-reach A,B,E so state ABE will be the start in the DFSM.

To get the 0-exit from ABE:
A -> none
B -> C
E -> none
Next state is C

To get the 1-exit from ABE:
A -> none
B -> C
E -> D -> (λ moves) B, D, E
Next state is BCDE

To get the 0-exit from C:
C -> D -> (λ moves) B, D, E
Next state is BDE

To get the 1-exit from C:
C -> none
Next state is ϕ

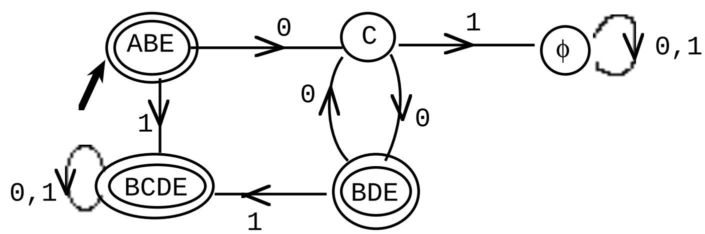
To get the 0-exit from BCDE:
B -> C
C -> D -> (λ moves) B, D, E
Next state is BCDE

To get the 1-exit from BCDE:
B -> C
C -> none
D -> none
E -> D -> (λ moves) B, D, E
Next state is BCDE

To get the 0-exit from BDE:
B -> C
D -> none
E -> none
Next state is C

To get the 1-exit from BDE:
B -> C
D -> none
E -> D -> (λ moves) B, D, E
Next state is BCDE

From state ϕ both exits go to next state ϕ



(d) The only state λ-reachable from C is C so the starting state in the DFSM will be named C.

To get the 0-exit from C:
C -> none
Next state is ϕ

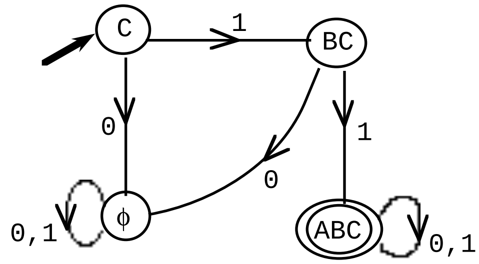
To get the 1-exit from C:
C -> B, C
Next state is BC

To get the 0-exit from BC:
B -> none
C -> none
Next state is ϕ

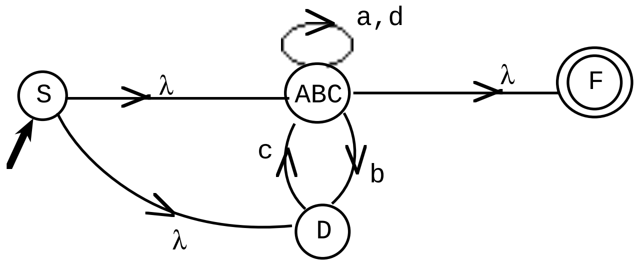
To get the 1-exit from BC:
B -> A
C -> B, C
Next state is ABC

To get the 0-exit from ABC:
A -> A, B, C
B -> doesn't matter
C -> doesn't matter
Next state is ABC

To get the 1-exit from ABC:
A -> none
B -> A
C -> B, C
Next state is ABC

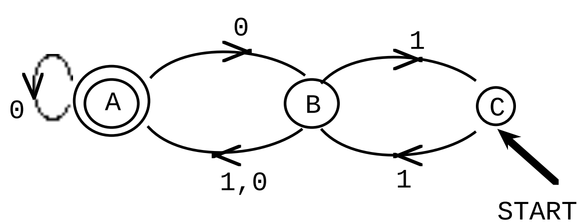


10. Path ABCA is a λ-cycle so each of A,B,C is λ-reachable from the others.
The three states can be merged into one state without changing the set of string recognized.



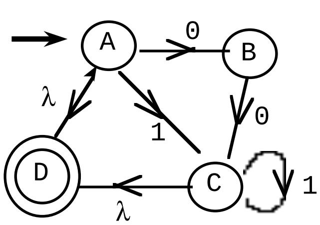
11. Make the old starting state accepting and no longer starting, make the old accepting state starting but no longer accepting, and reverse all transition arrows.

footnote This procedure is just for a FSM with *one* accepting state. If the FSM had more than one accepting state you could first convert to a FSM with just one accepting state (see problem 14) and then use this procedure.



12. (a) 2^{10} because there are 2^{10} subsets (including the empty subset) of a set with 10 elements
(b) 11. Keep all the original states and add the dead-end state ϕ.

13. (a) Fig B recognizes more than Fig A. For example, Fig B recognizes 100 and Fig A doesn't.
(b) Change all the accepting states to non-accepting and add λ-moves to a new accepting state. That way you get an equivalent NDFM with one accepting state (that turns out to not be the starting state).



SOLUTIONS Section 5.4

1. $A + B = \{ab, c, d, 0, 11\}$
   $AB = \{ab0, ab11, c0, c11, d0, d11\}$,
   $B^2 = \{00, 011, 110, 1111\}$
   To get B^*, juxtapose 0's and 11's; B^* contains λ, 000, $(11)^3\, 0^6 (11)^7\, 0$, 11 11, etc.

2. (a) all strings of 0's including the empty string
   (b) all non-empty strings of 0's
   (c) same as (b)
   (d) strings of all zeroes with an even number of 0's including the empty string λ
   (e) strings ending in 111
   (f) strings ending with 111 and containing no other 1's, i.e., strings of zeroes (maybe none of them) followed by 111
   (g) strings containing exactly two 1's
   (h) λ plus strings containing an even number (larger than zero) of 1's (i.e., except for λ, the strings must have two 1's or four 1's or six 1's etc.)
   For example, the set contains λ, 01001, 1111, 10101010 etc.

   (i) just the two strings 01 and 111

3. (a) Fig A (They have some strings in common but each contains strings that the other doesn't)
Region I contains abcabc, $(abc)^3$, $(abc)^4$ etc.
Region II contains just abc. Nothing else.
Region III contains λ, a,bc, a^2, $(bc)^2$, $a^2(bc)^2$, $a^5(bc)^7$ etc.

   (b) Fig D (Q contains everything that P does and then some).
   Region I (named $Q - P$) (what Q has that P doesn't) contains abc, bca, a^2bc, $(bc)^3a^3$ etc.
   Region II (named $P \cap Q$) (what they have in common) contains everything in Q, like a, bc, a^2, $(bc)^2$, a^3, $(bc)^3$ etc.

(c) Fig A Region I (named $P-Q$) contains abcabc, abcabcabc, etc
   Region II (named $P \cap Q$) contains just λ (each of P and Q contains λ and that's all they have in common)
   Region III (named $Q-P$) contains a, bc, a^2, $(bc)^2$, a^3, $(bc)^3$, etc.

4. (a) Yes. It contains λ^3 which is λ.

   (b) Yes. $01^3\,01^3 = \underbrace{01\;1^2\;01}_{\text{in } (1^*01)^*} \quad \underbrace{11}_{\text{in } 11\,+\,0^*}$

5. (a) $(a + bb)^* + b$ (b) doesn't simplify (c) $(01 + 1)^*$

6. (a) A^* (b) A^* (c) $(A + B)^*$ (d) A^*

7. (a) True
   (b) False
   (c) True. Each side concatenates strings in A and strings in B any which way.
   (d) False. It is true if A contains λ but if A doesn't contain λ then AA^* doesn't contain λ but A^* does and in that case $AA^* \neq A^*$.
   (e) True
   (f) False. If $A = \{10, 0, 111\}$ then $1A = \{110, 10, 1111\}$ and is not equal to A.
   (g) False. Here's a counterexample: $(0 + 1)^*$ does not equal $0^* + 1^*$
   $(0 + 1)^*$ contains *all* strings of 0's and 1's (including λ)
   $0^* + 1^*$ consists of strings of just 0's and strings of just 1's (and λ)

8. Answers aren't unique but disagree at your own risk.
(a) 0^*
(b) 0^*10^*
(c) $(0+1)^*1(0+1)^*$

(d) Add the answers to (a) and (b): $0^* + 0^*10^*$

(e) $0^*10^*10^*$

(f) $(0+1)^*1(0+1)^*1(0+1)^*$

(g) Add the answers to (a),(b),(e): $0^* + 0^*10^* + 0^*10^*10^*$

(h) Some possibilities are $0^*(10^*10^*)^*$, $(0^*10^*10^*)^* + 0^*$

(i) Begin and end with 1 or begin with 1 and end with 0
or begin and end with 0 or begin with 0 and end with 1.
An answer is $10(10)^*1 + 10(10)^* + 01(01)^*0 + (01)(01)^*$.

(j) $(0+1)^*1001\ (0+1)^*$

9. (a) abb^*a
(b) $a + b + c$
(c) $(a + b + c)^*cba(a + b + c)^*$
(d) $a + (bc)^*$
(e) $a(a + b + c)^*b$

(f) $(a + b + c)b(a + b + c)^*$

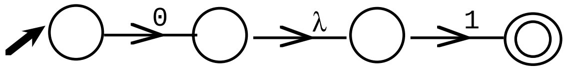
10. (a) $(a + bb)(a + bb)^*$
With $(a + bb)^*$, you have the option of picking nothing from a+ bb which is why the set contains λ. Sticking the extra factor a+bb in front has the effect of making all strings start with a or with bb, so that λ is no longer an option.
(b) $a + bc^* + \lambda$ (The definition of a regular expression does allow plus signs, meaning union.)

11. It comes pretty close but it misses λ,0,1 since it includes only strings of length *two* containing at most two 1's (you must pick something from each (0+1) factor. A correct answer is $0^*(0+1)0^*(0+1)0^* + \lambda + 0 + 1$. Another correct answer is in problem #8(g).

12. (a) right
(b) right
(c) wrong because (among other things) it misses 1
(d) wrong because it misses 101
(e) right.
(f) wrong because it misses 01101.

SOLUTIONS Section 5.5

1. (a) Start with the recognizers for 0 and 1 and hook them up in series to get a recognizer for 01.



Warning Kleene's FSM recognizer for 01 has a λ-move in the middle. Fig A below shows a (simpler) recognizer for 01 but it isn't Kleene's, i.e., it wasn't built using the standard procedure for getting a FSM for $r_1 r_2$.

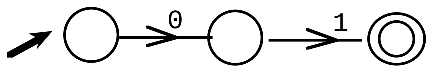
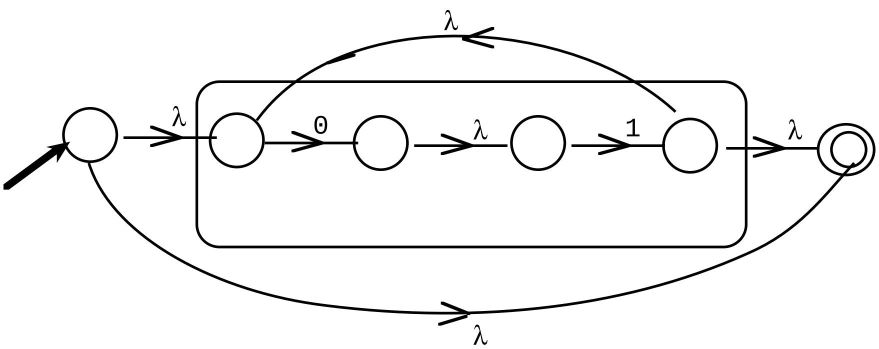
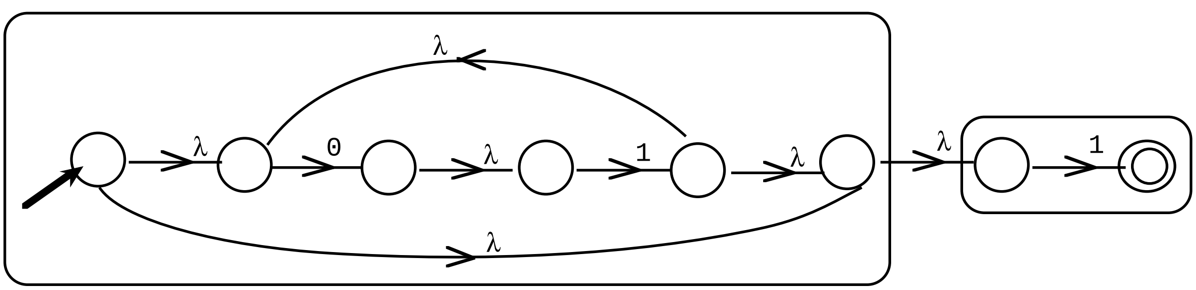


FIG A

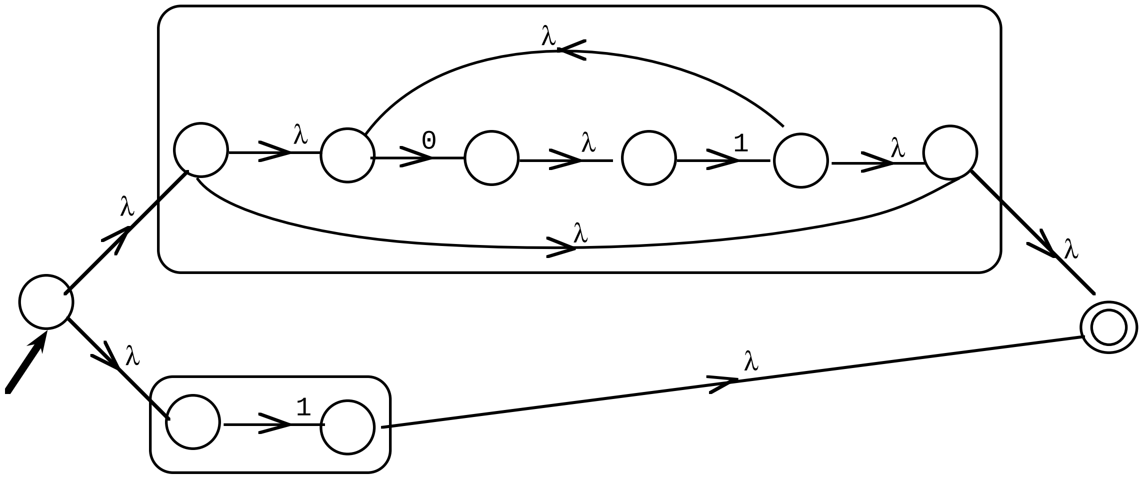
Then follow Kleene's procedure for getting $(01)^*$.



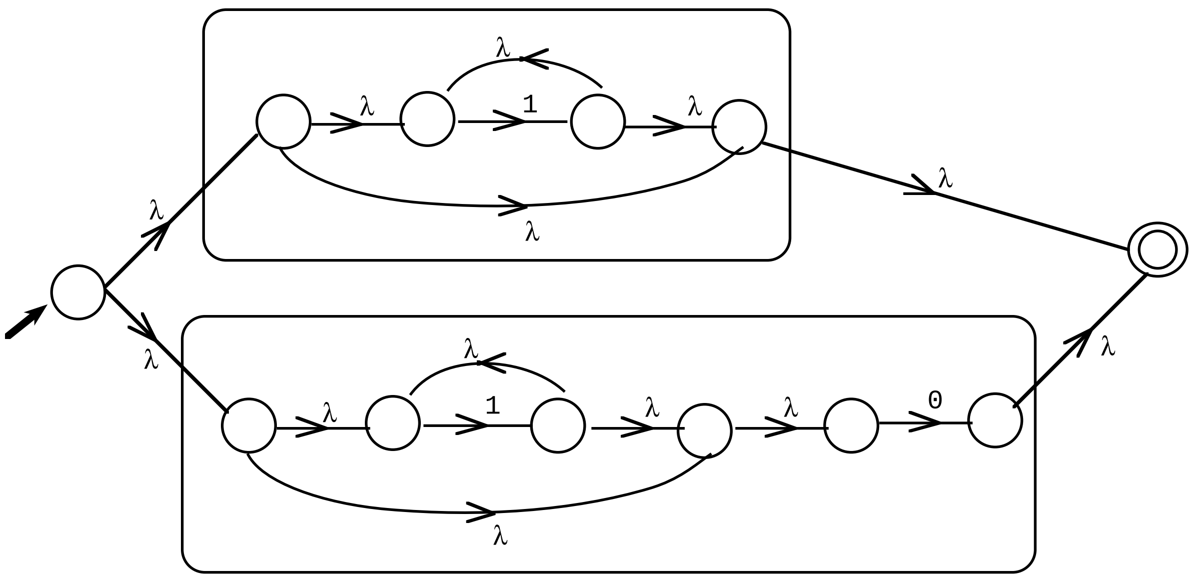
(b)



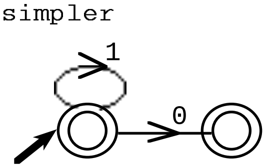
(c)



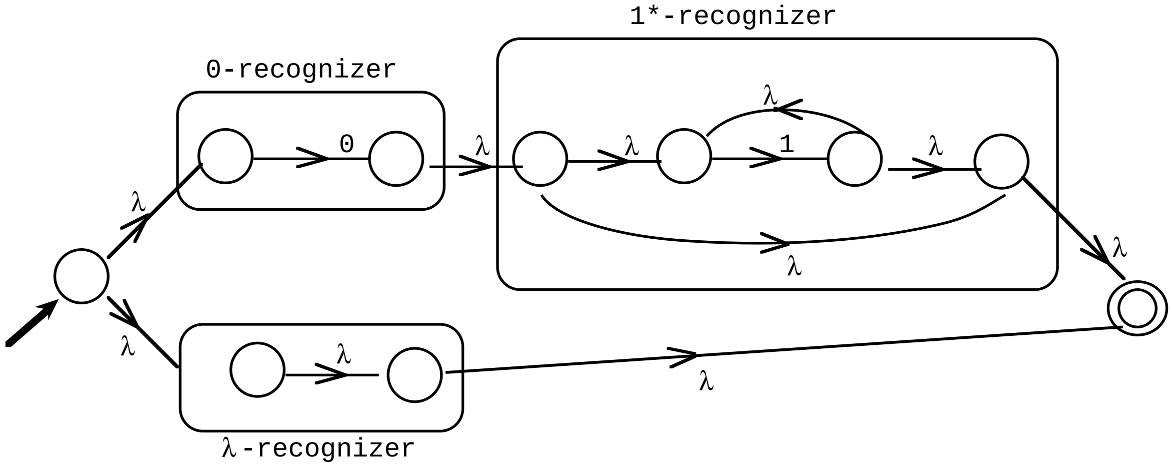
2. (a) (i) Kleene



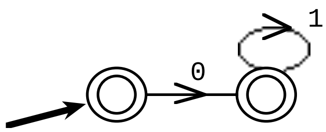
simpler



(ii) Kleene



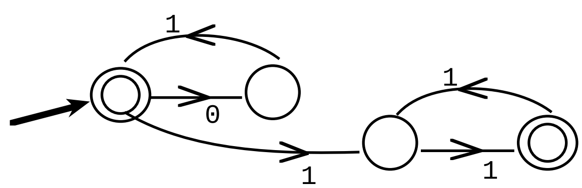
simpler



(b) The point of the Kleene construction is that it always works. It's a way of proving that it's *always* possible to get a FSM recognizer for a regular set.

3. (a)

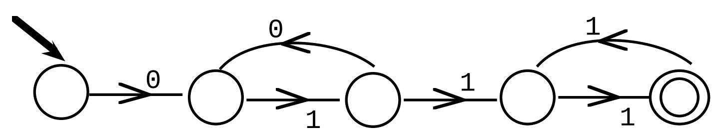
(01)*(11)*



Note that λ is in the set (let n = 0, m = 0) so the starting state must be accepting

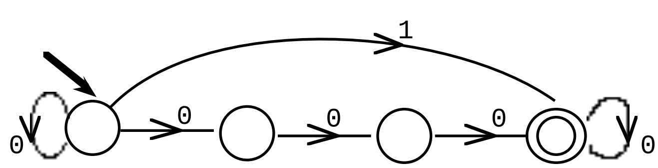
(b)

01(01)*11(11)*



(c)

0* 10* + 0000*



4. Doesn't work. Here's a counterexample.

Fig A shows a recognizer for $r_1 = a(ba)^*$ and Fig B shows a recognizer for $r_2 = c(dc)^*$. The proposed recognizer in Fig C for $r_1 r_2$ recognizes too much. It does recognize all the words in $r_1 r_2$ but it recognizes other words as well. For example, the string a c d b a c is recognized by Fig C but it is *not* in $r_1 r_2$.

You need that extra λ-move between the end of r_1 and the start of r_2 to prevent strings that have reached the r_2 recognizer from doubling back into the r_1 recognizer.

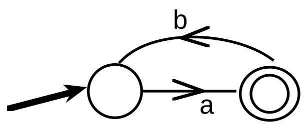


FIG A Recognizes $r_1 = a(ba)^*$

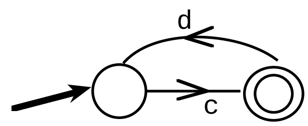


FIG B Recognizes $r_2 = c(dc)^*$

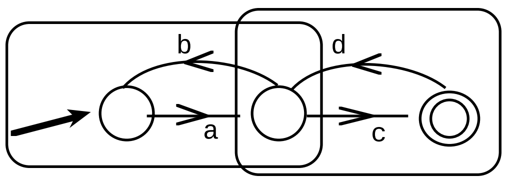


FIG C Proposed recognizer for r1 r2
NO GOOD

5. (a) Doesn't work.

It does work if λ is already in r. It does not work if λ is not in r.

Here's a specific counterexample.

Fig D shows a recognizer for 0.

Fig E is the proposed recognizer for 0^*. The proposed recognizer does not recognize λ. But it should because λ is in 0^*. Can't be right.

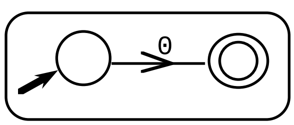


FIG D Recognizes 0

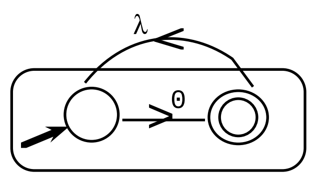


FIG E Proposed recognizer for 0^*
NO GOOD

(b) Doesn't work. All the strings in r^* including λ are recognized but it might recognize too much; i.e., it might recognize strings not in r^*.

Here are two counterexamples.

Counterexample 1 Fig F below shows a FSM which recognizes $r = 10^*$. The proposed recognizer for r^* is in Fig G.

r^* contains λ and strings like 111, 10011, 100100100. It does *not* contain 0 but Fig G unfortunately recognizes 0 (see path $A \xrightarrow{\lambda} B \xrightarrow{0} B$). Can't be right.

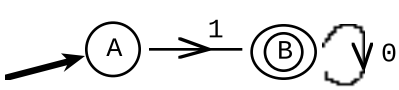


FIG F Recognizer for $r = 10^*$

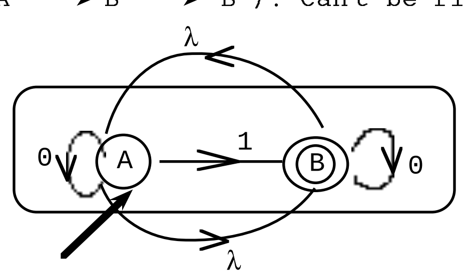


FIG G Proposed recognizer for r^*
NO GOOD

Counterexample 2 The FSM in Fig H below recognizes r = 0^*1. The proposed recognizer for r^* is in Fig I; it recognizes 0 (see path $A \xrightarrow{0} A \xrightarrow{\lambda} B$) but it shouldn't since 0 is not in r^*. Can't be right

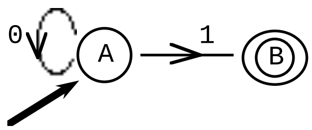


FIG H Recognizer for r = 0^*1

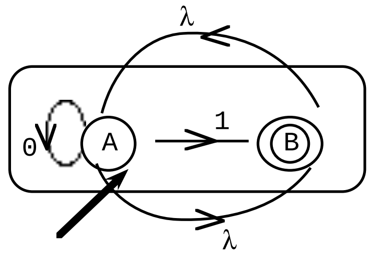


FIG I Proposed recognizer for r^*
NO GOOD

(c) Doesn't work. The proposed FSM might recognize too much.
Here's a counterexample. Look at the FSM in Fig J below which recognizes r = 10^*. The proposed recognizer for r^* is in Fig K; it recognizes 0 but it shouldn't since 0 is not in r^*.

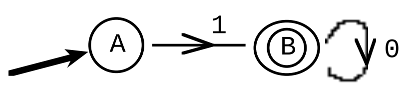


FIG J Recognizer for r = 10^*

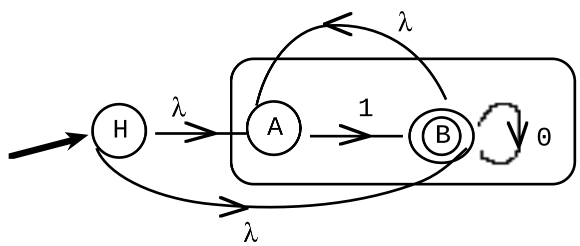


FIG K Proposed recognizer for r^*
NO GOOD

(d) Doesn't work. The proposed FSM might recognize too much.
Here's a counterexample. Look at the FSM in Fig L below which recognizes r = 0^*1. The proposed recognizer for r^* is in Fig M; it recognizes 0 but it shouldn't since 0 is not in r^*. So the proposal is no good.

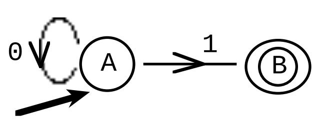


FIG L Recognizer for r = 0^*1

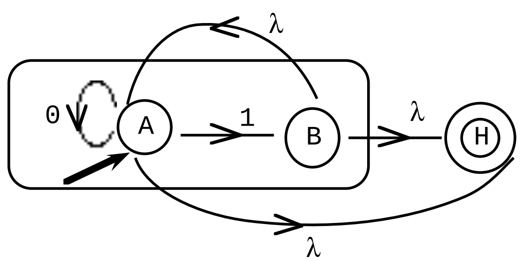
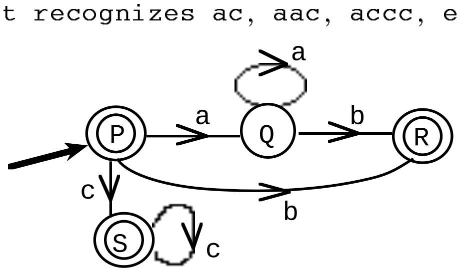


FIG M Proposed recognizer for r^* NO GOOD

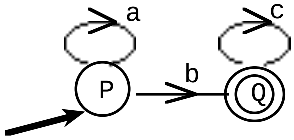
(e) Does work. (But it has two accepting states and for the proof of Kleene's theorem I was using λNFSMs with just one accepting state.)
(f) Does work.

6.(a) It recognizes ac, aac, accc, etc. which are *not* in $a^*b + c^*$

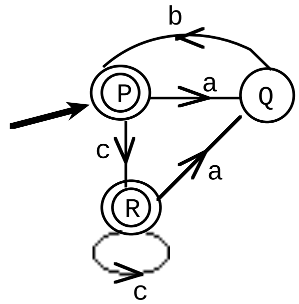
(b)



7. (a)

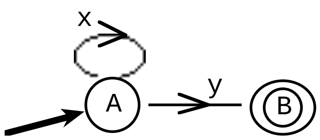


(b)

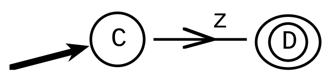


8. Doesn't work. Here is a counterexample.
Let r1 = x*y, r2 = z

Here are recognizers for r1 and r2.

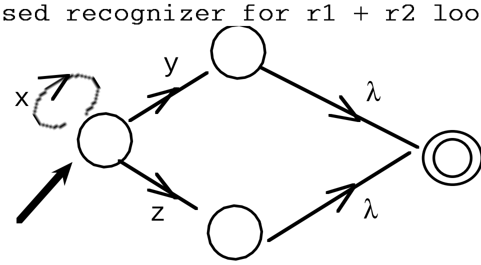


recognizer for r1



recognizer for r2

The proposed recognizer for r1 + r2 looks like this:



It recognizes more than r1 + r2. For instance it recognizes xz, which is not in r1 + r2.

You need the extra new start to prevent strings from dancing in one recognizer before going through the other recognizer.

SOLUTIONS Section 5.6

> Your answer doesn't have to agree with mine.
> A regular set can have more than one regular expression.

1. (a) $A = 0A + 0B + 1C + \lambda$
$B = 1A$
$C = \lambda$

So

$$A = 0A + 01A + 1\lambda + \lambda = (0 + 01)A + 1 + \lambda$$
$$A = (0 + 01)^*(1 + \lambda)$$

or equivalently $A = (0 + 01)^*1 + (0 + 01)^*$

check Words in $(0 + 01)^*1 + (0 + 01)^*$ are of the form
blocks of 0's & 01's (maybe none) maybe followed by 1

stay in A   cycle back to A   go to C

so they always end in an accepting state, either A or C.

(b) $A = 0B + 1A + \lambda$
$B = 1B + 1C$
$C = 1C + 1A + 0B$

So

$$C = 1^*(1A + 0B)$$
$$A = 0B + 1A + \lambda$$
$$B = 1B + 11^*(1A + 0B) = (1 + 11^*0)B + 11^*1A$$

and

$$A = 1A + 0(1 + 11^*0)^*\,11^*1A + \lambda = \left[1 + 0(1 + 11^*0)^*11^*1\right]A + \lambda$$

$$A = \left[1 + 0(1 + 11^*0)^*11^*1\right]^*\lambda = \left[1 + 0(1 + 11^*0)^*11^*1\right]^*$$

(c) $A = 1A + 0B$
$B = 1A + 0B + \lambda$

Then

$$B = 0^*(1A + \lambda)$$

$$A = 1A + 00^*(1A + \lambda) = (1 + 00^*1)A + 00^*$$

$$A = (1 + 00^*1)^*00^*$$

(Another answer is $A = 1^*0(11^*0)^*0^*$.)

(d) $A = 0B + 1C$
$B = 1B + 0F + \lambda$
$C = 1D$
$D = 0E$
$E = 1C + \lambda$
$F = 0F + 1F = (0 + 1)F$

$F = (0 + 1)^*\phi = \phi$
(obvious since starting from F,
a string is never accepted)

$B = 1B + \lambda$
$B = 1^*$
$C = 1D = 10E = 10(1C + \lambda) = 101C + 10$
$C = (101)^*10$
$A = 01^* + 1(101)^*10$

> **warning**
> The solution to
>
> $$F = (0 + 1)F$$
>
> is *not* $F = (0+1)^*$. Rather, it's
>
> $$F = (0+1)^*\phi = \phi$$

(e) $P = \lambda B$
$B = 1P + \lambda$

So

$B = 1^* \lambda = 1^*$
$P = \lambda 1^* = 1^*$

2. Can't decide which is the starting state. The starting state determines what letter you solve for but the state equations themselves are independent of which state is the starting state.

The only accepting state is A because A's equation is the only one with a λ term.

3. Try to find all the paths ending in an accepting state

(a) $1^* 000^*$

(b) $\underbrace{01^*}_{\text{end in B}} + \underbrace{110(110)^*}_{\text{end in E}}$

Another answer is $01^* + 1(101)^*10$

(c) The FSM now recognizes λ in addition to what it recognized before.
One method is to add λ to the answer to (b): answer is $01^* + 110(110)^* + \lambda$
Another possibility is $01^* + (110)^*$

(d) $11^*(01)^*$

(e) Two possibilities are $21(01)^*2 + 12^*1$ and $2(10)^*12 + 12^*1$

4. $(d + o + g + c + a + t)^*(dog + cat)\ (d + o + g + c + a + t)^*$
The FSM recognizes words containing dog or cat or both.

5. r_1, r_2, r_3 are regular because here are regular expressions for them:

$r_1 = (a+b+c+d)^*\ abc\ (a+b+c+d)^*$

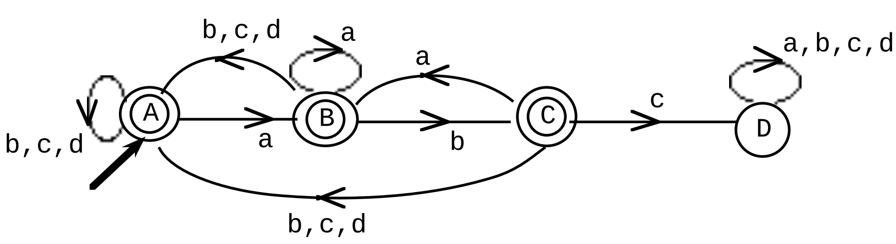
$r_2 = (a+b+c+d)^*\ abc\ (a+b+c+d)^*\ abc\ (a+b+c+d)^*$

$r_3 = (a+b+c+d)^*\ abc\ (a+b+c+d)^*\ abc\ (a+b+c+d)^*\ abc\ (a+b+c+d)^*$

Then $r_4 = \overline{r_1}$, $r_5 = r_1 - r_2$, $r_6 = r_2 - r_3$

So r_4, r_5, r_6 are regular.

Another method is to find FSM recognizers. The diagram shows one for r_4.



6. (a) Let

r_1 = set of strings containing 001

r_2 = set of strings with an even number of 1's

r_1 and r_2 are regular. One way to show it is to get regular expressions for them.

$r_1 = (0+1)^*\ 001\ (0+1)^*$

$r_2 = 0^*(10^*10^*)^*$

Then r is regular because $r = r_1 \cap r_2$ and the intersection of regular sets is regular.

(b) Let
r_1 = set of strings containing 001
r_3 = set of strings containing 0000
They are regular. I got r_1 reg in part (a) and $r_3 = (0+1)^*\ 0000\ (0+1)^*$.
Then s is regular because $s = r_1 - r_3$ (and it's a theorem that the difference of regular sets is regular).
(c) Let
r_4 = set of strings ending in 0010
Then r_4 is reg because $r_4 = (0+1)^*\ 0010$
And t is regular since $t = \overline{r_4}$ (and it's a theorem that the complement of a regular set is regular).

7. $r_1 - r_2 = r_1 \cap \overline{r_2}$. We already know that intersections and complements of regular sets are reg so if r_1 and r_2 are reg, so is $r_1 - r_2$. QED

8. *method 1* If r is reg then r has a FSM recognizer with just one accepting state (by Kleene part I). To get a recognizer for the set of reversed strings, make the old starting state accepting and no longer starting, make the old accepting state starting and no longer accepting, and reverse the direction of each transition

method 2 If r is reg. Then it has a reg expression. Then a reg expression can be found for s by reversing the order of all factors. So s is reg.
For example if $r = 012^*(a + b^*c)^* + d$ then $s = (a + cb^*)^*\ 2^*10 + d$

9. (a) Every regular set has a FSM recognizer (Kleene's theorem part I)
(b) The set of strings recognized by a FSM is regular (Kleene's theorem part II)

warning about language

Kleene's theorem Part II does *not* say that every FSM "describes" a regular set or "represents" a regular set or "expresses" a regular set" or even "has" a regular set. It says that every FSM *recognizes* (or *accepts*) a regular set.

The two theorems put together say that a set of strings is regular if and only if it has a FSM recognizer.

10. Step 1 Start with the regular expression for $\overline{r}$: $\overline{r} = (0 + 1)^*\ 1101\ (0 + 1)^*$.

Step 2 Get a FSM recognizer for $\overline{r}$ (guaranteed by Kleene, Section 5.5)

Step 3 Get a DFSM recognizer for $\overline{r}$
(§5.3, page 4, converting a NFSM to a DFSM)

Step 4 Use the procedure at the bottom of page 5 in Section 5.6 to get a FSM recognizer for the complement of $\overline{r}$, i.e., for r (change every acc state to non-acc and vice versa)

Step 5 Solve state equations to get a reg expression for the set recognized by the FSM in Step 4. QED

Question 1

The set r is $(0 + 1)^* - (0 + 1)^*\ 1101\ (0 + 1)^*$.
(Everything minus "at least one 1101")
So why isn't this a regular expression for r. Saves all those steps.

Answer

A regular expression is not allowed to contain a minus sign.

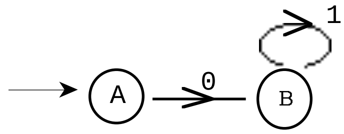
Question 2

The set r is $\overline{(0 + 1)^*\ 1101\ (0 + 1)^*}$

So why isn't this a regular expression for r. Saves all those steps.

Answer A reg express is not allowed to contain a complement symbol.

11.(a) You should get A = ϕ (the null set). When you leave out the λ your state equations actually go with the FSM below *which has no accepting states*. So of course it can't recognize any strings.



(b) Remember the algebra rules involving the null set ϕ:
For any set D,

(i) $D\phi = \phi$ (i.e., anything concatenated with ϕ is just ϕ again)
(ii) $D + \phi = D$ (i.e., union of any set with ϕ is just that set)

Start with
(3) $A = 0B$
(4) $B = 1B$

By (ii), equation (4) can be rewritten as

(5) $B = 1B + \phi$

Then
(6) $B = 1^*\phi$ use the P^*Q rule in (5) with $P = 1$, $Q = \phi$

(7) $B = \phi$ by (i) above

footnote
Of course the solution to (4) is $B = \phi$.
If B were anything else and you concatenated everything in B with 1, you certainly would not get B back again.
So the only B that satisfies $B = 1B$ is $B = \phi$

(8) $A = 0\phi$ substitute for B in (3)
(9) $A = \phi$ by (i) again

Which is what we predicted in part (a).
And not the answer you will get with the correct state equations.

(c) Remember the algebra rule involving the set λ (the set containing just the empty string λ)

(iii) $D\lambda = D$ for any set D

Start with
(1) $A = 0B$
(2) $B = 1B + \lambda$

Then
(10) $B = 1^*\lambda$ use the P^*Q rule in (2) with $P = 1$, $Q = \lambda$

(11) $B = 1^*$ by (iii)

(11) $A = 01^*$ substitute for B in (1)

Which is what I got by inspection.

Moral Don't forget the lambdas. You get wrong answers if you do.

SOLUTIONS Section 5.7

1. Suppose the set is regular.
Then it has a FSM recognizer. Let N be the number of states.
method 1 Look at the string $0^N 1^{N+1}$.
It has a track in the FSM leading to an accepting state.
Within the first N 0's there are N+1 state-boxes but only N states so there is a repeated state, i.e., there is a loop.

$$\underbrace{0 \quad \overbrace{\ldots}^{\text{loop}} \quad 0}_{\text{N zeroes, N+1 boxes}} \qquad \underbrace{1 \; \ldots \; 1}_{\text{N+1 ones}}$$

Make a new string by duplicating the block of 0's between the repeated states. The new string is still accepted (it reaches an accepting state along the original track but without the loop). But the new string doesn't have more 1's than 0's (at best it has an equal number of each) so it shouldn't be accepted. Contradiction!.
So the set is not regular.
(Note that if you *delete* the block of 0's, the new word still has more 1's than 0's and deserves to be accepted. No contradiction from this approach.)

method 2 Look at the same string, $0^N 1^{N+1}$. As the string goes on a track to an accepting state, there will be a repeated state (a loop) among the N+1 ones.

$$\underbrace{0 \quad \ldots \quad 0}_{\text{N zeroes}} \qquad \underbrace{1 \quad \overbrace{\ldots}^{\text{loop}} \quad 1}_{\text{N+1 ones}}$$

Make a new string by deleting the block of 1's corresponding to the loop. The new string is still accepted. But the new string no longer has more 1's than 0's so it shouldn't be accepted. Contradiction! So no such FSM exists and the set is not reg.
(Note that if you duplicate the block of 1's, the new word still has more 1's than 0's and deserves to be accepted. No contradiction from this approach)

2. (a) The string has a track leading to an accepting state.
Among the zeros in the first half of the word there will be a repeated state, i.e., a loop.

$$1 \quad \underbrace{\overbrace{0 \ldots 0}^{\text{N zeros}}}_{\text{loop in here}} \qquad 1 \quad \overbrace{0 \ldots 0}^{\text{N zeros}}$$

Consider the block of zeros corresponding to the loop (the block may contain as few as one 0 or as many as all N 0's). If you delete the block, the new string still has a track to an accepting state (the original track but without the loop). But it shouldn't: If the block had an odd number of 0's then the new string doesn't have an even number of symbols and doesn't even have halves. If the block had an even number of 0's then at least the new string still has halves but they don't match because the first half has two 1's and the second half is all 0's.
So the new string should not be recognized. Contradiction!
So the set is not regular

(b) There is a track leading to an accepting state. There will be a repeated state (a loop) along that track.

$$\underbrace{10\ldots0 \quad 10\ldots0}_{\text{loop somewhere in here}}$$

Consider the block of symbols corresponding to the loop. If it's deleted or duplicated, the new word is still accepted. But is that a contradiction?

Deleting it doesn't necessarily produce a contradiction because if the block were everything but the last two 0's then deleting it leaves the word 00 which *is* in the set and should be recognized.

Duplicating the block once (or any odd number of times) doesn't necessarily produce a contradiction either because if the block was the whole string then the new string still has repeated halves.

For example if your string was 100100 and the block was the whole string and you duplicated it once then the new string would be 100100 100100.Still has repeated halves.

Duplicating the block twice (or any even number of times) doesn't necessarily produce a contradiction either because if the block was the entire first half of the string then the new string still has repeated halves.

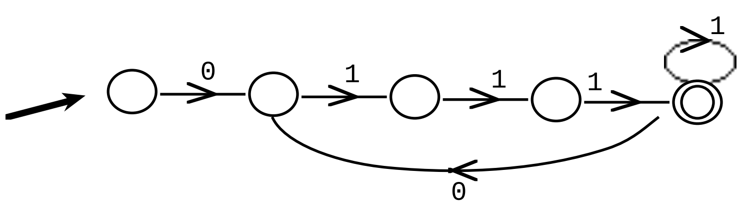
For example if your string was 100100 and the block was 100 and you duplicated it twice then the new string would be 100 100 100 100. Still has repeated halves.

So the argument doesn't go anywhere with this string.

Bottom Line The set is not regular but the string in (b) didn't help in the proof. The string in part (a) did make a proof work.

3. Whether you delete the block or duplicate once or duplicate q times, the new string still reaches an accepting state (skips the loop, goes around the loop an extra time, goes around the loop an extra q times).
But to get a contradiction you must be able to show that the new string doesn't deserve to be accepted, i.e., no longer has a prime number of zeros.
If the new string is obtained by deleting the block, it might still have a prime number of zeros (a smaller prime than q).
If the new string is obtained by duplicating the block one, it might still have a prime number of zeroes (a larger prime than q).
Duplicating the block q times was very clever. It definitely made the new string have a non-prime number of 0's. Hence the contradiction.

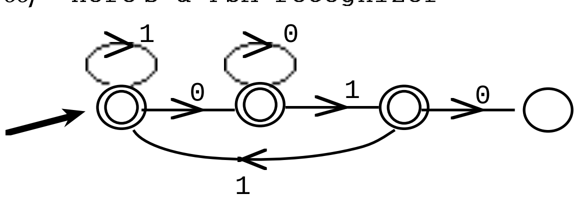
4. Regular
first proof The set has the reg expression 0 111 1* (0 111 1*)*
second proof Here is a FSM recognizer



5. Regular

first proof Let s be the set of strings that *do* contain at least one occurrence of 010. Then s is regular since s = (0+1)* 010 (0+1)*. The given set is $\overline{s}$ and it's regular since complements of reg sets are reg.

second proof Here's a FSM recognizer



6. Not reg. Here's the proof.
Suppose it is regular.
Then there is a FSM recognizer. Let N be the number of states.
Look at the string 0^{N^2}

$$\overbrace{0 \quad \underbrace{\ldots}_{\text{loop}} \quad 0}^{\text{first N zeroes}} \qquad \overbrace{0 \quad \ldots \quad 0}^{\text{rest of the zeroes (there are } N^2 - N \text{ of them)}}$$

There is a track leading to an accepting state. Among the first N zeroes there will be a repeated state in the track..
Look at the block of 0's corresponding to the loop. Let k be the number of 0's in the block. All we know about k is that $1 \le k \le N$.

method 1 Make a new string by deleting the block.
The new string still has a track to an accepting state (go on the original track but skip the loop).

claim The number of 0's in the new string is not a perfect square.

proof of claim

Here's an example first.
Suppose N = 3.
Then my original string is 0^9 and the block contains 1,2 or 3 0's.
The new string contains 8, 7 or 6 zeros. Not a perfect-square number of 0's

Here's the general argument.
The original string contains N^2 zeros.
The block contains 1 or 2 or ... or N zeros.
The number of zeros in the new string is between $N^2 - N = N(N-1)$ and $N^2 - 1$.
In particular, the smallest number of zeros possible in the new string is $N(N-1)$.
But $N(N-1)$ is still larger than the next lowest perfect square which is $(N-1)(N-1)$.
So the number of zeros in the new string can't be a perfect square (it's between the consecutive squares N^2 and $(N-1)^2$).
End of proof of claim.

So it is a contradiction that the new string reaches an accepting state.
So the set couldn't be regular.

method 2 It also works to make a new string by duplicating the block.
The new string still has a track to an accepting state (the original track but twice around the loop).

claim The number of 0's in the new string is not a perfect square.

proof of claim

The original string contains N^2 zeros.
The block contains 1 or 2 or ... or N zeros.
The number of zeros in the new string is between $N^2 + 1$ and $N^2 + N = N(N+1)$.
In particular, the largest number of zeros in the new string is $N(N+1)$.
But $N(N+1)$ is still smaller than the next highest perfect square which is $(N+1)(N+1)$.
So the number of zeros in the new string can't be a perfect square (it's between the consecutive squares N^2 and $(N+1)^2$).
End of proof of claim.

So it is a contradiction that the new string reaches an accepting state.
So there is no FSM recognizer and the set is not regular.

7. (a) The 1 that used to be in the middle is pushed out of the middle but a new 1 may take its place. This would happen for example if the first N symbols are all 1's.

The trouble is that this attempt is trying a string of the form $(0+1)^N\ 1\ (0+1)^N$. This is a whole collection of strings when you should try one specific string.

(b) Just be more specific and try the string $0^N\ 1\ 0^N$.

$$\underbrace{0\ ..\ 0}_{N\text{ zeros}}\ 1\ \underbrace{0\ ...\ 0}_{N\text{ zeros}}$$

The string has a track to an accepting state.
There is a loop within the first N zeros.
Duplicate twice the block of symbols corresponding to the loop.

The new word is still accepted but the 1 is now pushed out of the middle.

Note that if the block happens to contain just one zero and is deleted or duplicated once, the new word would have an odd number of symbols with two middle spots and there would still be a 1 in one of those middle spots. So to get my contradiction I duplicated twice to be sure. Duping ten times isn't necessary.

So it shouldn't be accepted. Contradiction.
So there is no FSM recognizer and the set is not regular.

8. Suppose the set of strings is regular.

Then it has a FSM recognizer. Let N be the number of states.

Pick the first string in the set with length $\geq$ N (there might not be one with length exactly N)

As the word goes on a track in the FSM to an accepting state there will be a repeated state, a loop.
Look at the block of symbols corresponding to the loop.

first try

Make a new string by deleting the block. The new string is still recognized.
But if the block is at the end then the new string might still be in the set and entitled to recognition. (For example, delete the 89 from 123456789 and the new string is still in the set.) So I don't have a contradiction. No conclusion yet.

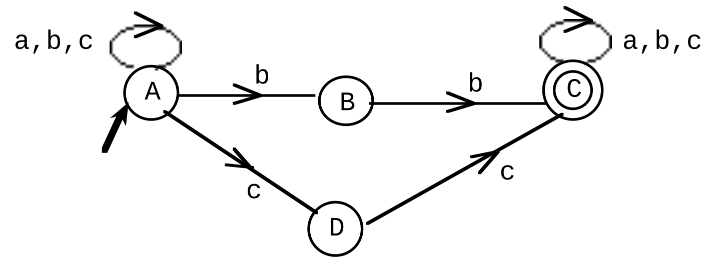
second try

Make a new string by duplicating the block. The new string is still accepted (go around the loop twice). But no matter what the block is and no matter where it is, the new string is not in the set and not entitled to recognition. Contradiction!

So there is no FSM recognizer and the set is not reg.

SOLUTIONS review problems for Chapter 5

1.



2. The state equations are

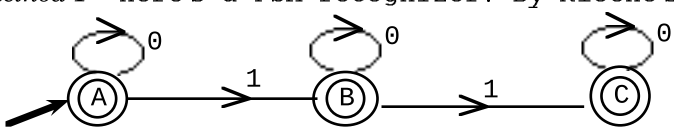
(1) A = 2B + 1D
(2) B = 1C
(3) C = 2E + 0B
(4) D = 2D + 1E
(5) E = λ

My plan is to solve for B and D and substitute into (1)

(6) D = 2*1 E (use P*Q rule in (4))
(7) D = 2*1 (since E = λ and Zλ = Z for any set Z)
(8) B = 1(2E + 0B) (substitute (3) into (2))
(9) B = 12 + 10B (algebra) (remember that E = λ)
(10) B = (10)*12 (use P*Q rule in (9))
(11) A = 2(10)*12 + 12*1 (substitute (10) and (7) into (1))

(Another answer is A = 21(01)*2 + 12*1.)

3. *method 1* Here's a FSM recognizer. By Kleene's theorem part II, that makes r reg.



method 2 Let s be the set of strings with at least three 1's. Then s is regular because

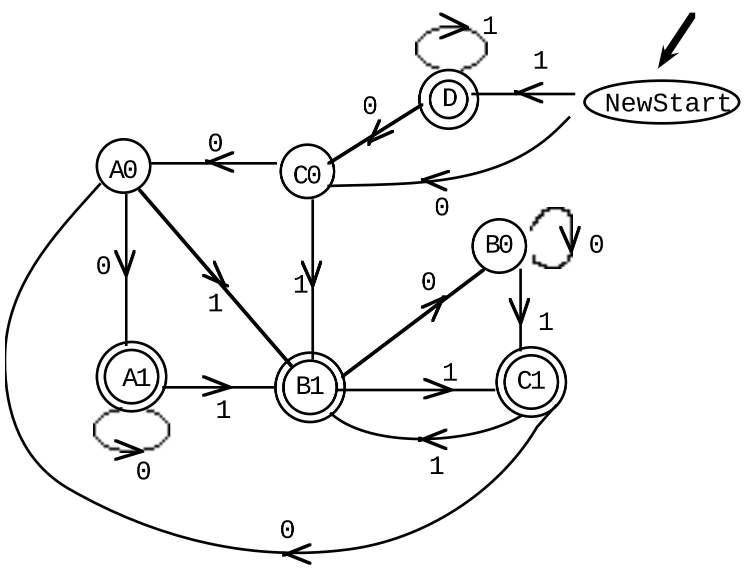
s = (0+1)* 1 (0+1)* 1 (0+1)* 1 (0+1)*

And r = $\overline{\text{s}}$ so r is regular

method 3

r = no 1's + one 1 + two 1's = 0* + 0* 1 0* + 0* 1 0* 1 0*

4.



5. $b^* a(a + ba)^* bb(a + b)^*$

6. (a) Strings of the form 01^n, $n \geq 1$, and strings of the form 11^n, $n \geq 1$.

(b) From A you can λ-reach A,B,D so the starting state in the DFSM is ABD.

To get the 0-exit from ABD
A -> none
B -> C -> (λ moves) D, F, P, Q, R
D -> none
Next state is CFPQR

To get the 1-exit from ABD
A -> none
B -> none
D -> E -> (λ) E, F, P, Q, R
Next state is EFPQR

To get the 0-exit from CFPQR
C, F, Q, P, R -> none
Next state is ϕ

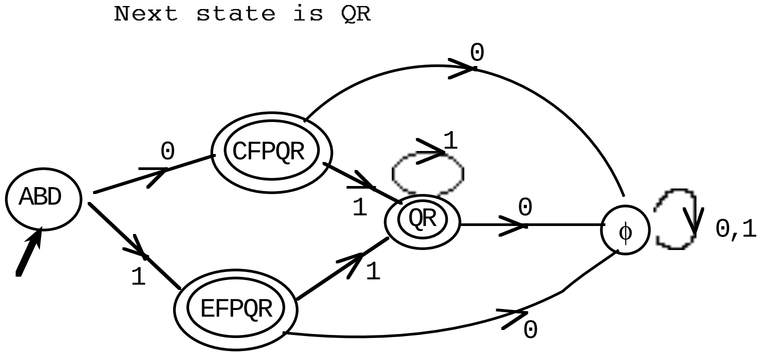
To get the 1-exit from CFPQR
Q -> R -> (λ) Q, R
C, F, P, R -> none
Next state is QR

To get the 0-exit from EFPQR
E, F, P, Q, R -> none
Next state is ϕ

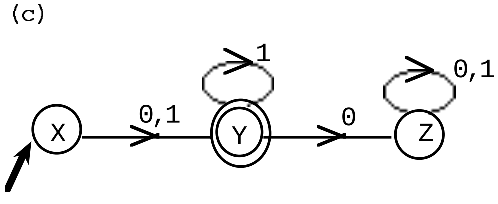
To get the 1-exit from EFPQR
Q -> R -> (λ) Q, R
E, F, P, R -> none
Next state is QR

To get the 0-exit from QR
Q, R -> none
Next state is ϕ

To get the 1-exit from QR
Q -> R ->(λ) Q, R
R -> none
Next state is QR



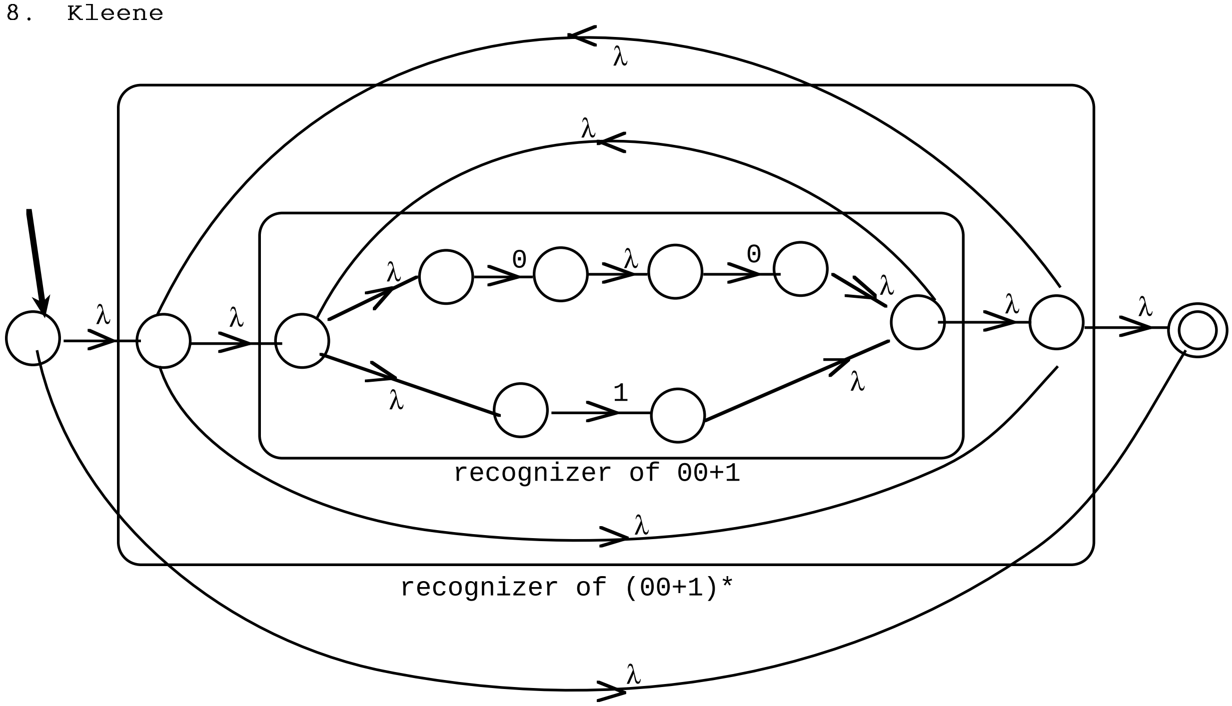
(c)



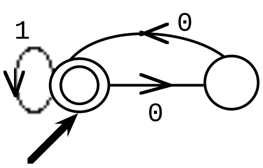
7.

in		λ's	1	λ's	0	λ's	
next	A	A,B,E	C,D	B,C,D,E	C,D	$\boxed{B}$,C,D,E	accepted

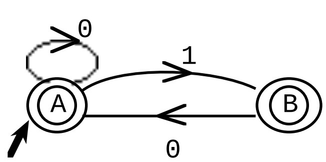
8. Kleene



simpler



9. $(0 + 10)^* (1 + \lambda)$



10. The string has a track leading to an accepting state.
There will be a repeated state somewhere in that track.

$$\underbrace{000\ 111\ 1^N}_{\text{loop in here}}$$

Make a new string by deleting the block between the repeated states.
The new string still has a track leading to an accepting state (the old track but without the loop) so the new string is still recognized.

If the block contains any of the three 0's, then the new string shouldn't be accepted and we do get a contradiction.
If the block contains at least N 1's then the new string ends up with fewer than four 1's and the new string shouldn't be accepted. Contradiction.
But if the block consists of N−1 or fewer 1's then the new string still starts

with three 0's followed by at least four 1's and it should be accepted. Not a contradiction.

So deleting the block doesn't *necessarily* get a contradiction.

So nothing has been proved.

This does not prove that the set is regular. It doesn't prove anything.

Suppose the block corresponding to the loop is *duplicated*.

Do you get a contradiction?

Yes if the block contains any 0's.

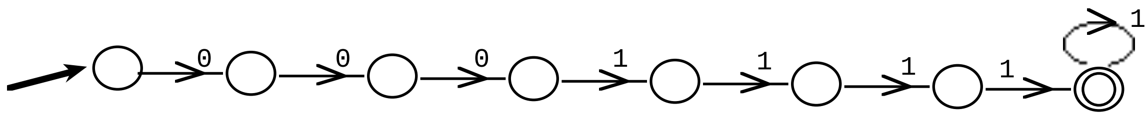
No if the block contains only 1's

Again, no conclusion.

As a matter of fact the set is regular since it has the regular expression

$000\,1111\;1^*$

You could also say that the set is reg since it has the following FSM recognizer.



11. $0(0 + 1)^*1$

(It's the set of strings that begin with a 0 and end with a 1.)

SOLUTIONS Section 6.1

1. *Part I* True for n = 1 (trivially) because LHS = 1 and RHS = $\frac{1}{2}\cdot 1\cdot(1+1) = 1$.

Part II Assume true for n=k; i.e., assume that for a particular k,

$$1 + 2 + 3 + \ldots + k = \frac{1}{2}k(k+1) \quad \text{(the induction hypothesis)}$$

Want to prove true for n=k+1, i.e.; want to prove

$$1 + 2 + 3 + \ldots + k+1 = \frac{1}{2}(k+1)(k+2).$$

Here's the proof:

$$1 + 2 + 3 + \ldots + k+1 = \underbrace{1 + 2 + 3 + \ldots + k}_{\frac{1}{2}k(k+1) \text{ by ind hyp}} + k+1$$

$$= = \frac{1}{2}k(k+1) + k+1$$

$$= (k+1)(\frac{1}{2}k + 1) \quad \text{factor}$$

$$= \frac{(k+1)(k+2)}{2} \quad \text{QED}$$

2. *Part I* Trivially true for n = 1 since LHS = $1^2 = 1$ and RHS = $\frac{1\cdot 2\cdot 3\cdot}{6} = 1$.

Part II Assume it's true for the particular value n=k; i.e., assume that for a particular k,

$$1^2 + 2^2 + \ldots + n^2 = \frac{n(n+1)(2n+1)}{6} \quad \text{(induction hypothesis)}$$

Want to prove true for n=k+1; i.e., want to prove

$$1^2 + 2^2 + \ldots + (k+1)^2 = \frac{(k+1)(k+2)(2[k+1] + 1)}{6} = \frac{(k+1)(k+2)(2k+3)}{6}$$

Here's the proof:

$$1^2 + 2^2 + \ldots + (k+1)^2 = 1^2 + 2^2 + \ldots + k^2 + (k+1)^2$$

$$= \frac{k(k+1)(2k+1)}{6} + (k+1)^2 \quad \text{by induction hypothesis}$$

$$= (k+1)\left[\frac{k(2k+1)}{6} + (k+1)\right] \quad \text{(factor)}$$

$$= (k+1)\frac{2k^2 + 7k + 6}{6} \quad \text{(algebra)}$$

$$= \frac{(k+1)(k+2)(2k+3)}{6} \quad \text{(factor)}$$

This proves part II.

3. *Part I* Obviously true for n = 1 since LHS = $1^3 = 1$ and RHS = $\left(\frac{1\cdot 2}{2}\right)^2 = 1$.

Part II Assume true for n=k; i.e., assume that for a particular k,

$$1^3 + 2^3 + \ldots + k^3 = \left(\frac{k(k+1)}{2}\right)^2 \quad \text{(the induction hypothesis)}$$

Want to prove that

$$1^3 + 2^3 + \ldots + (k+1)^3 = \left(\frac{(k+1)(k+2)}{2}\right)^2$$

proof:

$$1^3 + 2^3 + \dots + (k+1)^3 = 1^3 + 2^3 + \dots + k^3 + (k+1)^3$$

$$= \left[\frac{k(k+1)}{2}\right]^2 + (k+1)^3 \quad \text{(by ind hyp)}$$

$$= (k+1)^2 \left[\frac{k^2}{4} + (k+1)\right] \quad \text{(factor)}$$

$$= (k+1)^2 \, \frac{k^2 + 4k + 4}{4} \quad \text{(algebra)}$$

$$= (k+1)^2 \, \frac{(k+1)^2}{4} \quad \text{(algebra)}$$

$$= \left[\frac{(k+1)(k+2)}{2}\right]^2$$

4. *Part I* True for n = 1 since $11^1 - 4^1 = 7$ which *is* divisible by 7.
Part II Assume $11^k - 4^k$ is divisible by 7.
Try to show that $11^{k+1} - 4^{k+1}$ is divisible by 7.

Do some algebra to express $11^{k+1} - 4^{k+1}$ in terms of $11^k - 4^k$ so that you can use the induction hypothesis.

$$11^{k+1} - 4^{k+1} = 11 \cdot 11^k - 4 \cdot 4^k$$
$$= 11(11^k - 4^k) + 11 \cdot 4^k - 4 \cdot 4^k \quad \text{(rearrange)}$$
$$= 11\underbrace{(11^k - 4^k)}_{\substack{\text{div by 7} \\ \text{by ind hyp}}} + \underbrace{7 \cdot 4^k}_{\substack{\text{div} \\ \text{by 7}}}$$

So $11^{k+1} - 4^{k+1}$ is a sum of things divisible by 7 so it is divisible by 7. QED

5. *Part I* True for n = 1 because $2^{2 \cdot 1} - 1 = 2^2 - 1 = 3$ which is div by 3.
Part II Assume $2^{2k} - 1$ is divisible by 3 for some particular k.
Want to show that $2^{2(k+1)} - 1$ is div by 3, i.e., that $2^{2k+2} - 1$ is div by 3.
Do some algebra:

$$2^{2k+2} - 1 = 2^2 \cdot 2^{2k} - 1 = 4 \cdot 2^{2k} - 1 = 4(2^{2k} - 1) + 4 - 1 = 4\underbrace{(2^{2k} - 1)}_{\substack{\text{div by 3} \\ \text{by ind hyp}}} + \underbrace{3}_{\substack{\text{div} \\ \text{by 3}}}$$

So $2^{2k+2} - 1$ is div by 3. This proves Part II.

6. *Part I* True for n = 35 since you can pay with seven 5¢ stamps.
Part II Assume true if the postage is n=k (the induction hypothesis).
Try to prove true for postage n=k+1.

case 1 Suppose postage k can be paid entirely with 5¢ stamps. It takes at least 7 of them since $k \geq 35$. To pay for postage k+1, replace seven of the 5¢ stamps by four 9¢ stamps. This pays for postage k+1 using only 5's and 9's.

case 2 Suppose postage k is paid using at least one 9¢ stamp. To pay for postage k+1 in this case, replace a 9¢ stamp with two 5's. This pays for postage k+1.

That proves Part II.

7. *Part I* True for n = 1 since 2 > 1.

Part II Assume $2^k > k$ for a particular k. Want to show that $2^{k+1} > k+1$.

proof:

$$2^{k+1} = 2 \cdot 2^k > 2 \cdot k \quad (\text{since } 2^k > k \text{ by ind hypothesis})$$
$$= k + k \ge k + 1 \quad (\text{since } k \ge 1)$$

8. (a) Part I If n = 1 then $n^2 + 5n + 1$ is 7 which *is* odd.

Part II Assume that $k^2 + 5k + 1$ is odd (the induction hypothesis).

Want to show that $(k+1)^2 + 5(k+1) + 1$ is odd.

proof

$$(k+1)^2 + 5(k+1) + 1 = k^2 + 2k + 1 + 5k + 5 + 1 \quad (\text{multiply out})$$
$$= k^2 + 5k + 1 + 2k+6 \quad (\text{rearrange})$$
$$= \underbrace{k^2 + 5k + 1}_{\substack{\text{odd by} \\ \text{ind hyp}}} + \underbrace{2(k + 3)}_{\substack{\text{even because} \\ \text{of the 2}}}$$

= odd + even = even QED

(b) Part II Assume $k^2 + 5k + 1$ is even.

Want to show that $(k+1)^2 + 5(k+1) + 1$ is even.

proof

$$(k+1)^2 + 5(k+1) + 1 = k^2 + 2k + 1 + 5k + 5 + 1$$
$$= k^2 + 5k + 1 + 2k+6 \quad (\text{rearrange})$$
$$= \underbrace{k^2 + 5k + 1}_{\substack{\text{even by} \\ \text{ind hyp}}} + \underbrace{2(k + 3)}_{\substack{\text{even because} \\ \text{of the 2}}}$$

= even + even = even QED

So you can do Part II of the induction argument.

But you can't do any version of Part I (can't show $n^2 + 5n + 1$ is even for n = 1 or *any* "starting" value of n because by part (a) it is always odd).

9. *Part I* Obviously true for n = 1.

Part II Assume true for n=k; i.e.; assume that

$$(\cos\theta + i \sin\theta)^k = \cos k\theta + i \sin k\theta$$

Try to prove true for n=k+1; i.e., want to prove that

$$(\cos\theta + i \sin\theta)^{k+1} = \cos(k+1)\theta + i \sin(k+1)\theta$$

proof:

$$(\cos\theta + i \sin\theta)^{k+1}$$
$$= (\cos\theta + i \sin\theta)^k \cdot (\cos\theta + i \sin\theta) \quad (\text{algebra})$$
$$= (\cos k\theta + i \sin k\theta) \cdot (\cos\theta + i \sin\theta) \quad (\text{by ind hypothesis})$$
$$= \cos k\theta \cos\theta - \sin k\theta \sin\theta + i(\cos\theta \sin k\theta + \sin\theta \cos k\theta) \quad (\text{algebra})$$
$$= \cos(k\theta + \theta) + i \sin(\theta + k\theta) \quad (\text{trig identities})$$
$$= \cos(k+1)\theta + i \sin(k+1)\theta \quad (\text{factor})$$

10. *Part I* True for n = 1 since LHS = $\binom{1}{0} + \binom{1}{1} = 1 + 1 = 2$ and RHS = $2^1 = 2$.

Part II Assume true for n=k; i.e., assume $\binom{k}{0} + \binom{k}{1} + \dots + \binom{k}{k} = 2^k$.

Want to prove true for n=k+1; i.e., show that

(**) $\binom{k+1}{0} + \binom{k+1}{1} + \dots + \binom{k+1}{k+1} = 2^{k+1}$

Here's the proof (which uses Pascal's identity over and over again):

LHS of (**)

$$= \underbrace{\binom{k+1}{0}}_{\text{same as } \binom{k}{0}} + \underbrace{\binom{k+1}{1}}_{\boxed{\binom{k}{0}} + \binom{k}{1} \text{ by Pascal}} + \underbrace{\binom{k+1}{2}}_{\boxed{\binom{k}{1}} + \binom{k}{2}} + \underbrace{\binom{k+1}{3}}_{\boxed{\binom{k}{2}} + \binom{k}{3}} + \dots + \underbrace{\binom{k+1}{k}}_{\boxed{\binom{k}{k-1}} + \binom{k}{k}} + \underbrace{\binom{k+1}{k+1}}_{\text{same as } \boxed{\binom{k}{k}}}$$

The boxed terms add up to 2^k by the induction hypothesis.

Similarly the unboxed terms add up to 2^k too.

So the LHS of (**) is $2^k + 2^k = 2 \cdot 2^k = 2^{k+1}$. QED

SOLUTIONS Section 6.2

1. *Part I* Show true for n = 1.

$$\left(\frac{1+\sqrt{5}}{2}\right)^0 = 1, \quad F_1 = 1. \text{ So } F_1 \le \left(\frac{1+\sqrt{5}}{2}\right)^0 .$$

And show true for n = 2.

$$\left(\frac{1+\sqrt{5}}{2}\right)^1 \ge 1 \text{ by inspection (because } \sqrt{5} \ge 1)$$
$$= 1$$

and $F_2 = 1$ so $F_2 \le \left(\frac{1+\sqrt{5}}{2}\right)^1$.

Part II Assume that (*) is true for two particular consecutive values n=k−1 and n=k; i.e., assume

$$F_{k-1} \le \left(\frac{1+\sqrt{5}}{2}\right)^{k-2} \quad \text{and} \quad F_k \le \left(\frac{1+\sqrt{5}}{2}\right)^{k-1} \qquad \text{(the induction hypothesis)}$$

Try to prove (*) for n=k+1; i.e., prove that $F_{k+1} \le \left(\frac{1+\sqrt{5}}{2}\right)^k$.

proof

$F_{k+1} = F_k + F_{k-1}$ (by definition)

$$\le \left[\frac{1+\sqrt{5}}{2}\right]^{k-1} + \left[\frac{1+\sqrt{5}}{2}\right]^{k-2} \qquad \text{(by the induction hypothesis)}$$

$$= \left[\frac{1+\sqrt{5}}{2}\right]^{k-2} \left[\frac{1+\sqrt{5}}{2} + 1\right] \qquad \text{(factor)}$$

$$= \left[\frac{1+\sqrt{5}}{2}\right]^{k-2} \left[\frac{1+\sqrt{5}}{2}\right]^2 \qquad \text{(by the given identity)}$$

$$= \left[\frac{1+\sqrt{5}}{2}\right]^{k} \qquad \text{QED (end of part II)}$$

2. (a) *Part I* True for n = 2, because 2 is prime.

Part II Assume true for n=k; i.e., assume k is either prime or factors into primes.

Try to prove true for n=k+1, i.e., either k+1 is prime or factors into primes.

proof: If k+1 is prime you're finished.

Suppose k+1 isn't prime. You want to show that it factors into primes.

Since k+1 isn't prime it factors into pq where $2 \le p \le k$ and $2 \le q \le k$.

But you're not finished because p and q aren't necessarily primes. The induction hypothesis says that k is prime or factors into primes but you don't know about p and q. So you're stuck.

(b) *Part I* Same as in (a).

Part II Assume true for 2,...,k; i.e., assume every number from 2 through k is either prime or factors into primes. This is the strong induction hypothesis.

Try to prove true for n=k+1; i.e., try to show that k+1 is either prime or factors into primes.

As above, if k+1 is prime you're finished.

So consider the case that k+1 isn't prime. Then it factors into pq where

$2 \le p \le k$ and $2 \le q \le k$. By the *strong* induction hypothesis p and q are themselves

prime or else factor into primes. So k+1 factors into primes which proves part II.

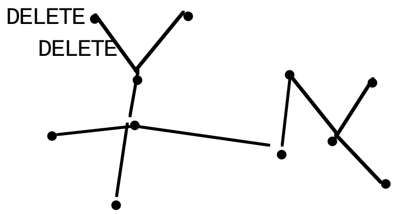
Then, by the principle of strong induction, every integer ≥ 2 is either prime or can be factored into a product of primes.

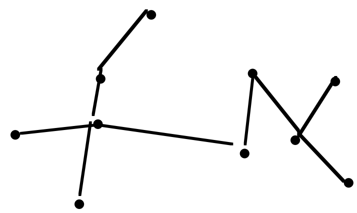
3. (a) Show that $V = E+1$ for a tree with 1 edge.
A tree with 1 edge has to look like this.

In this case $V = 2$ so V does equal $E + 1$.

Actually you could start by showing that it is true for a tree with zero edges. If $E = 0$ then the tree consists of just a single vertex. Then $E = 0$, $V = 1$ and V does equal $E+1$.

(b) Start with a tree with k+1 edges. Delete an "outer" edge and the dangling vertex at its end like this:





new graph

The new graph is still a tree but it has one less edge, i.e., $E_{new} = k$.
So by the induction hypothesis, in the new tree

$$(*) \qquad V_{new} = E_{new} + 1.$$

The original tree and the new tree are related like this:

$$V_{new} = V_{orig} - 1 \text{ and } E_{new} = E_{orig} - 1.$$

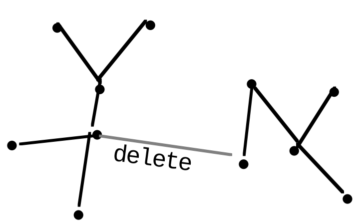
\
Substitute these into (*):

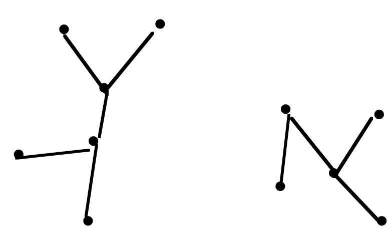
$$V_{orig} - 1 = E_{orig} - 1 + 1$$

And cancel the −1's to get

$$V_{orig} = E_{orig} + 1 \quad \text{QED}$$

(c) Start with a tree with $E = k+1$ edges and V vertices and delete an edge (no vertices are deleted). You get two new trees; call them Tree 1 and Tree 2





Tree 1 Tree 2

Let the number of vertices and edges in Tree 1 be denoted by V_1 and E_1.
Let the number of vertices and edges in Tree 2 be denoted by V_2 and E_2.
Each of Trees 1 and 2 has k *or fewer* edges (since we deleted one of the original k+1 edges and split the rest into two trees) so you can use the *strong* induction hypothesis on each of Tree 1 and Tree 2 (if your induction hypothesis had just been

that a tree with k edges has V = E + 1 then you would be stuck here):

$$V_1 = E_1 + 1$$
$$V_2 = E_2 + 1$$

Add to get

(**) $$V_1 + V_2 = E_1 + E_2 + 2$$

Here's how the new trees are related to the original tree with V vertices and E edges:

(*) $V_1 + V_2 = V$ and $E_1 + E_2 = E-1$ (remember you lost an edge)

Substitute (*) into (**):

$$\underbrace{V_1 + V_2}_{V} = \underbrace{E_1 + E_2}_{E-1} + 2$$

which becomes

$$V = E + 1 \quad \text{QED}$$

SOLUTIONS Section 7.1

1. (a) Antireflexive because no one is her own descendent.
Antisymmetric because if John is a descendent of Mary then Mary is certainly not a descendent of John
transitive (If John is a desc of Mary and Mary is a desc of Harry then John is a desc of Harry)
(b) reflexive, symmetric, transitive
(c) symmetric
Not reflexive because 3 $\overline{R}$ 3 for instance.
Not transitive because 3 R 4 and 4 R 5 but 3 $\overline{R}$ 5 (3·4 is even and 4·5 is even but 3·5 is not even]
(d) symmetric
Not reflexive because 7 $\overline{R}$ 7 for instance.
Not transitive because 6 R 3 and 3 R 5 but 6 $\overline{R}$ 5
(6 + 3 < 10 and 3 + 5 < 10 but 6 + 5 is not < 10).
(e) antireflexive, antisymm, trans
(f) antiref, symm.
Not trans because if L_1 and L_2 are two different parallel lines then L_1 R L_2 and L_2 R L_1 but L_1 $\overline{R}$ L_1.
(g) reflexive, symm, trans
(h) antireflexive, symm
Not trans because if $x \perp y$ and $y \perp z$ then x and z are parallel or coincident, not perp.

2. antireflexive, antisymm
Not trans because rock R scissors and scissors R paper but rock notR paper

3. (a) not reflexive Missing (2,2) (among others)
not antireflexive Does contain (1,1)
not symm Missing (1,2)
antisymm (2,1) is the only pair you care about, and it does not reverse.
trans The only hookup is 2 R 1 and 1 R 1 so you need 2 R 1 for transitivity and you have it
(b) not reflexive. Missing (2,2)
not antireflexive. Contains (1,1)
symm
antisymm by default There are no pairs (x,y) where $x \neq y$ to consider reversing

(c) not reflexive. Missing (2,2)
antireflexive
not symm Missing (3,2) for instance
antisymm
not transitive. Has (4,2) and (2,3) but not (4,3)

4. There are 4 pairs in A × A, namely (a,a), (b,b), (a,b), (b,a). And a set with 4 elements has 2^4 subsets. So there are 16 relations on A.

Equivalently use (a,a), (b,b), (a,b), (b,a) as slots and fill each slot with either YES (in the relation) or NO (not in the relation. So there are 2^4 possibilities. Here they are.
$R_1 = \phi$ (the empty relation where no one is related to anyone else)
antiref, symm, antisymm, trans (the last three, by default)

$R_2 = \{ (a,a) \}$ symm, antisymm, trans
$R_3 = \{ (b,b) \}$ same properties as R_2
$R_4 = \{ (a,a), (b,b) \}$ reflexive, symm, antisymm, trans
$R_5 = \{ (a,b) \}$ antiref, antisymm, trans
$R_6 = \{ (b,a) \}$ same properties as as R_5

R_7 = { (a,b), (b,a) } symm, antiref

Not trans because you have the hookup aRb, bRa but don't have aRa

R_8 = { (a,a), (a,b)} antisymm, trans

Trans because the only hookup is aRa, aRb and you do also have aRb

R_9 = { (a,a), (b,a) } Same properties as R_8

R_{10} = {(b,b), (a,b) } antisymm, trans

Trans because the only hookup is aRb, bRb and you do have aRb

R_{11} = {(b,b), (b,a) } Same as R_{10}

R_{12} = {(a,a),(b,b),(a,b)} reflex, antisymm, trans

Here's why R_{12} is trans: There are two hookups to check.

The first one is aRb, bRb and you do have aRb
The second one is aRa, aRb and you do have aRb

R_{13} = {(a,a), (b,b), (b,a)} reflex, antisymm, trans (just like R_{12})

R_{14} = {(a,a), (a,b), (b,a)} symm

Not trans because you have the hookup bRa, aRb but not bRb

R_{15} = {(b,b), (a,b), (b,a) } Same as R_{14}

$R_{16} = A \times A$ = {(a,a), (b,b), (a,b), (b,a)} ref, symm, trans

5. (a) There are $5 \cdot 5 = 25$ ordered pairs of elements in U. Think of each pair as a slot which can be filled in two ways: Related or Unrelated (i.e, in R or not in R). So there are 2^{25} relations.

Equivalently, there are $5 \cdot 5 = 25$ pairs in $A \times A$. There are 2^{25} subsets of $A \times A$ so there are 2^{25} relations.

(b) The relation must contain the five pairs $(x_1, x_1), \ldots, (x_5, x_5)$. The other 20 pairs are slots that can be filled in two ways: In the relation R or not in. Answer is 2^{20}.

(c) The relation must contain the six pairs $(x_1, x_1), \ldots, (x_5, x_5)$, (x_2, x_4). The other 19 pairs are slots that can be filled in two ways: Related or Unrelated. Answer is 2^{19}.

(d) There are 5 pairs of the form (x_i, x_i). Each is a slot which can be filled in two ways: YES or NO (in the relation vs. out of the relation) (you have this choice because a symmetric relation doesn't care whether or not $x_1 R\, x_1$).

There are $5 \cdot 4$ pairs of the form (x_i, x_j) where $i \neq j$.
Imagine them paired up so that you have 10 pairs of matching pairs, e.g., (x_1, x_3) & (x_3, x_1), (x_3, x_5) & (x_5, x_3) etc.

For each of these 10 pairs of pairs you have two choices: BOTH IN or BOTH OUT; e.g., (x_1, x_3) and (x_3, x_1) are either both in the relation or both out.

Answer is $2^5 \cdot 2^{10}$

(e) This is like part (d) but now all 5 pairs of the form (x_i, x_i) must be included in the relation. Less choice than in part (d). The answer is 2^{10}.

6. (a) It's really a plain graph in the sense that each edge is two-way.
(b) There's a loop at each vertex.
(c) If there is a path from v to w then there is also an edge directly from v to w (e.g., if v R p, p R q, q R z, z R w then v R w)

7. (a) Yes
Here is a relation defined on the universe of integers that is antisymmetric and also symmetric: x R y if x = y (where x and y are integers)
For instance 3 R 3, 4 R 4, 5 R 5 etc.
It's symmetric because if x R y (meaning x = y) then y R x.
And it's antisymm by default since you never have x R y where x ≠ y to begin with.
Alternatively, it is antisymm because if x ≠ y then we have neither x R y nor y R x.

Another example. Suppose the universe is {a, b, c}.
And the relation is { (a, a), (b, b) }, i.e., a R a and b R b.
Then it is symmetric (each pairing reverses).
And also antisymm because for two different x and y, it is true that either x R y or y R x or neither but not both. In particular, here you always have NEITHER.

(b) No. If the universe U contains x then a reflexive relation on U must have x R x and an antireflexive relation must *not* have x R x so a relation can't do both (unless the universe is the empty set to begin with but that would be a trick question).

(c) Yes
For example let the universe be {a, b, c} and let the relation be {(a, b), (b, a), (a, c) }.
The relation is not symmetric because (c, a) is missing.
And it is not antisymmetric because both (a, b) and (b, a) are in the relation.

For example, suppose the universe is the set of all people and x R y means x likes y.
Somewhere in the world there are pairs where the liking goes just one way, e.g., where John likes Mary but Mary doesn't like John. So the relation is not symmetric.
And somewhere in the world there are pairs where the liking goes both ways (e.g., Mary likes Sue and Sue likes Mary). So the relation is not antisymmetric.

(d) Yes. For example if the universe is {1, 2, 3, 4} and the relation is { (2, 2), (3, 4) } then it is not reflexive because R does not contain (3,3). And it not antireflexive because it does contain (2,2).
For example if x R y means that x has cooked a meal for y then R is not reflexive (there are some people who have never cooked meals for themselves) and it is not antireflexive (some people *have* cooked for themselves)

SOLUTIONS Section 7.2

1. (a) Yes. There is an equivalence class for each name. For instance there is a John Smith equiv class containing all the people named John Smith; there's a Mary Brown equiv class containing the people named Mary Brown etc.
(b) Yes There is a chemistry equiv class containing all the chem majors etc. There are as many equiv classes as there are subjects chosen as majors.
(c) No It's reflexive and symm but not transitive. If John has class with Mary and Mary has a class with Bill then John does not necessarily have a class with Bill.
(d) Yes There are $26 \cdot 26$ equivalence classes, one for people initialed A.A., another for people initialed A.B.,..., another for people initialed Z.Z. (unless there is a set of initials like X.Z. which no one has in which case there is no equivalence class named X.Z. ----it is *not* the case that there is a class but it's empty).
(e) Yes There are 50 equiv classes, one containing all the Florida residents, one containing all the Illinois residents etc.
(f) No It's reflexive and symmetric but not transitive.
(g) Yes. Each equivalence class contains exactly one person since no two people have the same SS number. There are as many classes as there are SS numbers.
(h) Yes Each equiv class contains exactly one integer and there are as many equiv classes as there are integers.
(i) Yes There are 26^4 equiv classes, one containing words of the form abcz..., one containing words of the form aaaa..., etc.
(j) Yes Two points are equiv if they are the same height above the x-axis. The equiv classes are the lines parallel to the x-axis. For example, the equiv class of the point (3,4) is the line $y = 4$.
(k) Yes The traditional equivalence classes are married, divorced, widowed, never married.
(ℓ) No Not reflexive (you aren't married to yourself) and it isn't transitive (if A is married to B and B is married to A then it isn't true that A is married to A, and if Mary is married to John-the-Bigamist and John is married to Susan then Mary is not married to Susan).

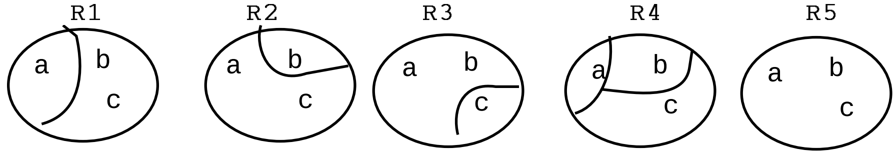
2. (a) R is an equivalence relation because it's symmetric, reflexive and transitive:
Reflexive $n\ R\ n$ for all n because $\sin n\pi = \sin n\pi$
Symm If $n\ R\ m$ then $m\ R\ n$ because if $\sin n\pi = \sin m\pi$ then $\sin m\pi = \sin n\pi$
Trans If $n\ R\ m$ and $m\ R\ k$ then $n\ R\ k$ because
if $\sin n\pi = \sin m\pi$ and $\sin m\pi = \sin k\pi$ then $\sin n\pi = \sin k\pi$.

There is only one equivalence class, namely $\{0, 1, 2, 3, 4, 5, \ldots\}$ because $\sin n\pi = \sin m\pi$ (= 0) for all integers n, m.

(b) R is an equivalence relation. There are two equivalence classes, $\{0, 2, 4, 6, \ldots\}$ and $\{1, 3, 5, 7, \ldots\}$
because $\cos 0 = \cos 2\pi = \cos 4\pi = \cos 6\pi$ etc (they all equal 1)
and $\cos \pi = \cos 3\pi = \cos 5\pi$ etc (they all equal -1).

3. There are 5 equiv relations

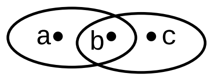
R1 R2 R3 R4 R5



For instance R1 = { (a,a), (b,b), (c,c), (b,c), (c,b) }
i.e., a R1 a, b R1 b, c R1 c, b R1 c, c R1 b.

R4 = { (a,a), (b,b), (c,c) }, i.e., a R4 a, b R4 b, c R4 c.

4. If two equivalence classes did overlap as in the diagram below then you would have a ~ b (because a and b are in the same cell), b ~ c (because b and c are in the same cell) but a not ≈ c (because a and c are not in the same cell). But this contradicts transitivity. So overlapping equivalence classes can't happen.



5. (a) Must prove that R is symmetric and transitive (already have reflexive).

symmetry

Suppose x R y. We want to show that y R x.
We have x R x by reflexivity.
But now that we have x R x and x R y we also have y R x by circularity, QED

transitivity

Suppose x R y and y R z. We want to show that x R z

From x R y and y R z we have z R x by circularity.
Then x R z by the symmetry that we just proved. QED

(b) Suppose R is an equivalence relation. We want to show that R is circular.
Suppose a R b and b R c. We want to show that c R a.
We have a R c by transitivity and then c R a by symmetry. QED