

## 14.8 The Pigeonhole Principle

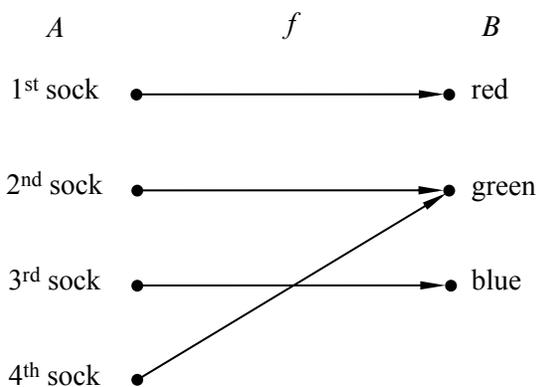
Here is an old puzzle:

A drawer in a dark room contains red socks, green socks, and blue socks. How many socks must you withdraw to be sure that you have a matching pair?

For example, picking out three socks is not enough; you might end up with one red, one green, and one blue. The solution relies on the

### **Pigeonhole Principle**

*If there are more pigeons than holes they occupy, then at least two pigeons must be in the same hole.*



**Figure 14.3** One possible mapping of four socks to three colors.

What pigeons have to do with selecting footwear under poor lighting conditions may not be immediately obvious, but if we let socks be pigeons and the colors be three pigeonholes, then as soon as you pick four socks, there are bound to be two in the same hole, that is, with the same color. So four socks are enough to ensure a matched pair. For example, one possible mapping of four socks to three colors is shown in Figure 14.3.

A rigorous statement of the Principle goes this way:

**Rule 14.8.1** (Pigeonhole Principle). *If  $|A| > |B|$ , then for every total function  $f : A \rightarrow B$ , there exist two different elements of  $A$  that are mapped by  $f$  to the same element of  $B$ .*

Stating the Principle this way may be less intuitive, but it should now sound familiar: it is simply the contrapositive of the Mapping Rules injective case (4.6). Here, the pigeons form set  $A$ , the pigeonholes are the set  $B$ , and  $f$  describes which hole each pigeon occupies.

Mathematicians have come up with many ingenious applications for the pigeonhole principle. If there were a cookbook procedure for generating such arguments, we’d give it to you. Unfortunately, there isn’t one. One helpful tip, though: when you try to solve a problem with the pigeonhole principle, the key is to clearly identify three things:

1. The set  $A$  (the pigeons).
2. The set  $B$  (the pigeonholes).
3. The function  $f$  (the rule for assigning pigeons to pigeonholes).

### 14.8.1 Hairs on Heads

There are a number of generalizations of the pigeonhole principle. For example:

**Rule 14.8.2** (Generalized Pigeonhole Principle). *If  $|A| > k \cdot |B|$ , then every total function  $f : A \rightarrow B$  maps at least  $k + 1$  different elements of  $A$  to the same element of  $B$ .*

For example, if you pick two people at random, surely they are extremely unlikely to have *exactly* the same number of hairs on their heads. However, in the remarkable city of Boston, Massachusetts, there is a group of *three* people who have exactly the same number of hairs! Of course, there are many completely bald people in Boston, and they all have zero hairs. But we’re talking about non-bald people; say a person is non-bald if they have at least ten thousand hairs on their head.

Boston has about 500,000 non-bald people, and the number of hairs on a person’s head is at most 200,000. Let  $A$  be the set of non-bald people in Boston, let  $B = \{10,000, 10,001, \dots, 200,000\}$ , and let  $f$  map a person to the number of hairs on his or her head. Since  $|A| > 2|B|$ , the Generalized Pigeonhole Principle implies that at least three people have exactly the same number of hairs. We don’t know who they are, but we know they exist!

### 14.8.2 Subsets with the Same Sum

For your reading pleasure, we have displayed ninety 25-digit numbers in Figure 14.4. Are there two different subsets of these 25-digit numbers that have the same sum? For example, maybe the sum of the last ten numbers in the first column is equal to the sum of the first eleven numbers in the second column?

Finding two subsets with the same sum may seem like a silly puzzle, but solving these sorts of problems turns out to be useful in diverse applications such as finding good ways to fit packages into shipping containers and decoding secret messages.

It turns out that it is hard to find different subsets with the same sum, which is why this problem arises in cryptography. But it is easy to prove that two such subsets *exist*. That’s where the Pigeonhole Principle comes in.

Let  $A$  be the collection of all subsets of the 90 numbers in the list. Now the sum of any subset of numbers is at most  $90 \cdot 10^{25}$ , since there are only 90 numbers and every 25-digit number is less than  $10^{25}$ . So let  $B$  be the set of integers  $\{0, 1, \dots, 90 \cdot 10^{25}\}$ , and let  $f$  map each subset of numbers (in  $A$ ) to its sum (in  $B$ ).

We proved that an  $n$ -element set has  $2^n$  different subsets in Section 14.2. Therefore:

$$|A| = 2^{90} \geq 1.237 \times 10^{27}$$

14.8. The Pigeonhole Principle

575

0020480135385502964448038	3171004832173501394113017
5763257331083479647409398	8247331000042995311646021
0489445991866915676240992	3208234421597368647019265
5800949123548989122628663	8496243997123475922766310
1082662032430379651370981	3437254656355157864869113
6042900801199280218026001	8518399140676002660747477
1178480894769706178994993	3574883393058653923711365
6116171789137737896701405	8543691283470191452333763
1253127351683239693851327	3644909946040480189969149
6144868973001582369723512	8675309258374137092461352
1301505129234077811069011	3790044132737084094417246
6247314593851169234746152	8694321112363996867296665
1311567111143866433882194	3870332127437971355322815
6814428944266874963488274	8772321203608477245851154
1470029452721203587686214	4080505804577801451363100
6870852945543886849147881	8791422161722582546341091
1578271047286257499433886	4167283461025702348124920
6914955508120950093732397	9062628024592126283973285
1638243921852176243192354	4235996831123777788211249
6949632451365987152423541	9137845566925526349897794
1763580219131985963102365	4670939445749439042111220
7128211143613619828415650	9153762966803189291934419
1826227795601842231029694	4815379351865384279613427
7173920083651862307925394	9270880194077636406984249
1843971862675102037201420	4837052948212922604442190
7215654874211755676220587	9324301480722103490379204
2396951193722134526177237	5106389423855018550671530
7256932847164391040233050	9436090832146695147140581
2781394568268599801096354	5142368192004769218069910
7332822657075235431620317	9475308159734538249013238
2796605196713610405408019	5181234096130144084041856
7426441829541573444964139	9492376623917486974923202
2931016394761975263190347	5198267398125617994391348
7632198126531809327186321	9511972558779880288252979
2933458058294405155197296	5317592940316231219758372
7712154432211912882310511	9602413424619187112552264
3075514410490975920315348	5384358126771794128356947
7858918664240262356610010	9631217114906129219461111
8149436716871371161932035	3157693105325111284321993
3111474985252793452860017	5439211712248901995423441
7898156786763212963178679	9908189853102753335981319
3145621587936120118438701	5610379826092838192760458
8147591017037573337848616	9913237476341764299813987
3148901255628881103198549	5632317555465228677676044
5692168374637019617423712	8176063831682536571306791

**Figure 14.4** Ninety 25-digit numbers. Can you find two different subsets of these numbers that have the same sum?

On the other hand:

$$|B| = 90 \cdot 10^{25} + 1 \leq 0.901 \times 10^{27}.$$

Both quantities are enormous, but  $|A|$  is a bit greater than  $|B|$ . This means that  $f$  maps at least two elements of  $A$  to the same element of  $B$ . In other words, by the Pigeonhole Principle, two different subsets must have the same sum!

Notice that this proof gives no indication *which* two sets of numbers have the same sum. This frustrating variety of argument is called a *nonconstructive proof*.

#### The \$100 prize for two same-sum subsets

To see if it was possible to actually *find* two different subsets of the ninety 25-digit numbers with the same sum, we offered a \$100 prize to the first student who did it. We didn't expect to have to pay off this bet, but we underestimated the ingenuity and initiative of the students. One computer science major wrote a program that cleverly searched only among a reasonably small set of “plausible” sets, sorted them by their sums, and actually found a couple with the same sum. He won the prize. A few days later, a math major figured out how to reformulate the sum problem as a “lattice basis reduction” problem; then he found a software package implementing an efficient basis reduction procedure, and using it, he very quickly found lots of pairs of subsets with the same sum. He didn't win the prize, but he got a standing ovation from the class—staff included.

#### The \$500 Prize for Sets with Distinct Subset Sums

How can we construct a set of  $n$  positive integers such that all its subsets have *distinct* sums? One way is to use powers of two:

$$\{1, 2, 4, 8, 16\}$$

This approach is so natural that one suspects all other such sets must involve larger numbers. (For example, we could safely replace 16 by 17, but not by 15.) Remarkably, there are examples involving *smaller* numbers. Here is one:

$$\{6, 9, 11, 12, 13\}$$

One of the top mathematicians of the Twentieth Century, Paul Erdős, conjectured in 1931 that there are no such sets involving *significantly* smaller numbers. More precisely, he conjectured that the largest number in such a set must be greater than  $c2^n$  for some constant  $c > 0$ . He offered \$500 to anyone who could prove or disprove his conjecture, but the problem remains unsolved.

### 14.8.3 A Magic Trick

A Magician sends an Assistant into the audience with a deck of 52 cards while the Magician looks away.

Five audience members each select one card from the deck. The Assistant then gathers up the five cards and holds up four of them so the Magician can see them. The Magician concentrates for a short time and then correctly names the secret, fifth card!

Since we don't really believe the Magician can read minds, we know the Assistant has somehow communicated the secret card to the Magician. Real Magicians and Assistants are not to be trusted, so we expect that the Assistant would secretly signal the Magician with coded phrases or body language, but for this trick they don't have to cheat. In fact, the Magician and Assistant could be kept out of sight of each other while some audience member holds up the 4 cards designated by the Assistant for the Magician to see.

Of course, without cheating, there is still an obvious way the Assistant can communicate to the Magician: he can choose any of the  $4! = 24$  permutations of the 4 cards as the order in which to hold up the cards. However, this alone won't quite work: there are 48 cards remaining in the deck, so the Assistant doesn't have enough choices of orders to indicate exactly what the secret card is (though he could narrow it down to two cards).

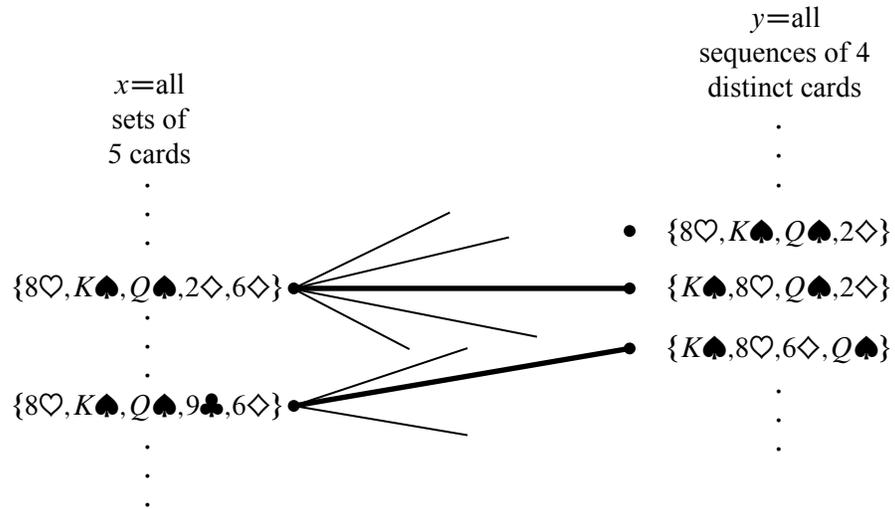
### 14.8.4 The Secret

The method the Assistant can use to communicate the fifth card exactly is a nice application of what we know about counting and matching.

The Assistant has a second legitimate way to communicate: he can choose *which of the five cards to keep hidden*. Of course, it's not clear how the Magician could determine which of these five possibilities the Assistant selected by looking at the four visible cards, but there is a way, as we'll now explain.

The problem facing the Magician and Assistant is actually a bipartite matching problem. Each vertex on the left will correspond to the information available to the Assistant, namely, a *set* of 5 cards. So the set  $X$  of left hand vertices will have  $\binom{52}{5}$  elements.

Each vertex on the right will correspond to the information available to the Magician, namely, a *sequence* of 4 distinct cards. So the set  $Y$  of right hand vertices will have  $52 \cdot 51 \cdot 50 \cdot 49$  elements. When the audience selects a set of 5 cards, then the Assistant must reveal a sequence of 4 cards from that hand. This constraint is represented by having an edge between a set of 5 cards on the left and a sequence of 4 cards on the right precisely when every card in the sequence is also in the set. This specifies the bipartite graph. Some edges are shown in the diagram in



**Figure 14.5** The bipartite graph where the nodes on the left correspond to *sets* of 5 cards and the nodes on the right correspond to *sequences* of 4 cards. There is an edge between a set and a sequence whenever all the cards in the sequence are contained in the set.

Figure 14.5.

For example,

$$\{8♥, K♠, Q♠, 2♦, 6♦\} \tag{14.2}$$

is an element of  $X$  on the left. If the audience selects this set of 5 cards, then there are many different 4-card sequences on the right in set  $Y$  that the Assistant could choose to reveal, including  $(8♥, K♠, Q♠, 2♦)$ ,  $(K♠, 8♥, Q♠, 2♦)$ , and  $(K♠, 8♥, 6♦, Q♠)$ .

What the Magician and his Assistant need to perform the trick is a *matching* for the  $X$  vertices. If they agree in advance on some matching, then when the audience selects a set of 5 cards, the Assistant reveals the matching sequence of 4 cards. The Magician uses the matching to find the audience’s chosen set of 5 cards, and so he can name the one not already revealed.

For example, suppose the Assistant and Magician agree on a matching containing the two bold edges in Figure 14.5. If the audience selects the set

$$\{8♥, K♠, Q♠, 9♣, 6♦\}, \tag{14.3}$$

then the Assistant reveals the corresponding sequence

$$(K♠, 8♥, 6♦, Q♠). \tag{14.4}$$

Using the matching, the Magician sees that the hand (14.3) is matched to the sequence (14.4), so he can name the one card in the corresponding set not already revealed, namely, the 9♣. Notice that the fact that the sets are *matched*, that is, that different sets are paired with *distinct* sequences, is essential. For example, if the audience picked the previous hand (14.2), it would be possible for the Assistant to reveal the same sequence (14.4), but he better not do that; if he did, then the Magician would have no way to tell if the remaining card was the 9♣ or the 2♦.

So how can we be sure the needed matching can be found? The answer is that each vertex on the left has degree  $5 \cdot 4! = 120$ , since there are five ways to select the card kept secret and there are  $4!$  permutations of the remaining 4 cards. In addition, each vertex on the right has degree 48, since there are 48 possibilities for the fifth card. So this graph is *degree-constrained* according to Definition 11.5.5, and so has a matching by Theorem 11.5.6.

In fact, this reasoning shows that the Magician could still pull off the trick if 120 cards were left instead of 48, that is, the trick would work with a deck as large as 124 different cards—without any magic!

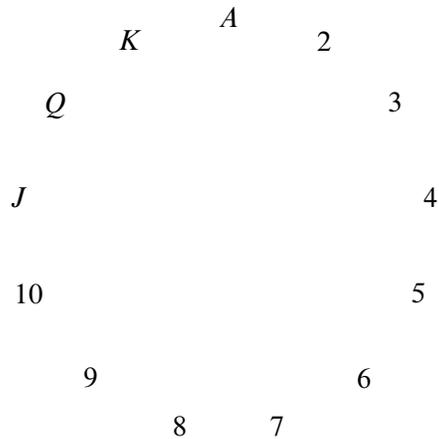
### 14.8.5 The Real Secret

But wait a minute! It’s all very well in principle to have the Magician and his Assistant agree on a matching, but how are they supposed to remember a matching with  $\binom{52}{5} = 2,598,960$  edges? For the trick to work in practice, there has to be a way to match hands and card sequences mentally and on the fly.

We’ll describe one approach. As a running example, suppose that the audience selects:

10♥ 9♦ 3♥ Q♠ J♦.

- The Assistant picks out two cards of the same suit. In the example, the assistant might choose the 3♥ and 10♥. This is always possible because of the Pigeonhole Principle—there are five cards and 4 suits so two cards must be in the same suit.
- The Assistant locates the ranks of these two cards on the cycle shown in Figure 14.6. For any two distinct ranks on this cycle, one is always between 1 and 6 hops clockwise from the other. For example, the 3♥ is 6 hops clockwise from the 10♥.
- The more counterclockwise of these two cards is revealed first, and the other becomes the secret card. Thus, in our example, the 10♥ would be revealed, and the 3♥ would be the secret card. Therefore:



**Figure 14.6** The 13 card ranks arranged in cyclic order.

- The suit of the secret card is the same as the suit of the first card revealed.
- The rank of the secret card is between 1 and 6 hops clockwise from the rank of the first card revealed.
- All that remains is to communicate a number between 1 and 6. The Magician and Assistant agree beforehand on an ordering of all the cards in the deck from smallest to largest such as:

$A\clubsuit A\diamond A\heartsuit A\spadesuit 2\clubsuit 2\diamond 2\heartsuit 2\spadesuit \dots K\heartsuit K\spadesuit$

The order in which the last three cards are revealed communicates the number according to the following scheme:

- ( small, medium, large ) = 1
- ( small, large, medium ) = 2
- ( medium, small, large ) = 3
- ( medium, large, small ) = 4
- ( large, small, medium ) = 5
- ( large, medium, small ) = 6

In the example, the Assistant wants to send 6 and so reveals the remaining three cards in large, medium, small order. Here is the complete sequence that the Magician sees:

$10\heartsuit Q\spadesuit J\diamond 9\diamond$

- The Magician starts with the first card,  $10\heartsuit$ , and hops 6 ranks clockwise to reach  $3\heartsuit$ , which is the secret card!

So that’s how the trick can work with a standard deck of 52 cards. On the other hand, Hall’s Theorem implies that the Magician and Assistant can *in principle* perform the trick with a deck of up to 124 cards. It turns out that there is a method which they could actually learn to use with a reasonable amount of practice for a 124-card deck, but we won’t explain it here.

### 14.8.6 The Same Trick with Four Cards?

Suppose that the audience selects only *four* cards and the Assistant reveals a sequence of *three* to the Magician. Can the Magician determine the fourth card?

Let  $X$  be all the sets of four cards that the audience might select, and let  $Y$  be all the sequences of three cards that the Assistant might reveal. Now, on one hand, we have

$$|X| = \binom{52}{4} = 270,725$$

by the Subset Rule. On the other hand, we have

$$|Y| = 52 \cdot 51 \cdot 50 = 132,600$$

by the Generalized Product Rule. Thus, by the Pigeonhole Principle, the Assistant must reveal the *same* sequence of three cards for at least

$$\left\lceil \frac{270,725}{132,600} \right\rceil = 3$$

*different* four-card hands. This is bad news for the Magician: if he sees that sequence of three, then there are at least three possibilities for the fourth card which he cannot distinguish. So there is no legitimate way for the Assistant to communicate exactly what the fourth card is!

---

MIT OpenCourseWare  
<https://ocw.mit.edu>

6.042J / 18.062J Mathematics for Computer Science  
Spring 2015

For information about citing these materials or our Terms of Use, visit: <https://ocw.mit.edu/terms>.