The Division Algorithm

E. L. Lady

(July 11, 2000)

Theorem [Division Algorithm]. Given any strictly positive integer d and any integer a, there exist unique integers q and r such that

$$a = qd + r,$$

and

$$0 \leq r < d\,.$$

Before discussing the proof, I want to make some general remarks about what this theorem really says, why it says it in what seems at first such a perversely obscure way, and why it's worth proving something at all which, as we shall see, actually seems quite obvious.

The Language in Which Mathematics Is Done

Even when talking about day to day life, it sometimes happens that one has a hard time finding the words to express one's thoughts. Furthermore, even when one believes that one has been totally clear, it sometimes happens that there is a vagueness in one's words that results in the listener understanding something completely different from what the speaker intended.

Even the simplest mathematical reasoning is usually more complicated than the most complicated things one attempts to discuss in everyday life. For this reason, mathematics has developed an extremely specialized (and often stylized) language. To see the value of this, one only has to think of how difficult it would be to solve even the simple quadratic equation $x^2 - 5x = 14$ if one's calculation had to be done without symbols, using only ordinary language.

> "Suppose that a quantity has the property that when five times that quantity is subtracted from the product of the quantity times itself then the result is 14. Determine the possible values of this quantity."

During the middle ages, algebraic calculations were in fact done in this fashion — except that the language was Latin, not English. But progress beyond the level of simple algebra only became possible with the introduction of modern algebraic notation — which the West learned from the Arabs.

By combining symbolic notation with very precisely defined concepts and a very formalized language, any mathematical proof can be expressed in a way that can be understood by any "mathematically mature" reader. One is not dependent on the reader's ability to "see what is meant." In principle, at least, every step is described in terms so precise that no possible misunderstanding can occur, and anyone can follow the reasoning from one step to the next, provided only that he is skilled in making the most basic sort of algebraic, set theoretic, and logical manipulations. In particular, the language is so explicit that one never needs reference to illustrative examples in order to explain the meaning of what is said. (In practice, it has to be admitted that sometimes even the experienced mathematician gets stopped at certain points in the proof and has to rack his brains for a minute or two —or even longer— before he says, "Oh, yeah, that's why." Students, of course, have this difficulty much more frequently.)

It's important to realize, however, that useful as the language of mathematics is, it doesn't represent the way humans normally gain understanding. So even something quite familiar can sometimes seem strange when described in a precise, mathematical way.

One might think of the language in which mathematics is done as analogous to programming languages. A computer program describes how to perform a given task without omitting even a single step, no matter how small, and without any conceivable ambiguity. One might think that for this reason, a computer program would be the ideal way to explain a procedure so that no one could possibly fail to understand it. But in fact, human beings find computer programs extremely difficult to understand.

The language of mathematical proofs is a little more human than that of computer programs. Nonetheless, learning to read this kind of language is a formidible challenge for students. Learning to write it is even more formidible. The endeavor can be quite worthwhile, however, since the reward for success is access to the vast realm of modern mathematics.

With this in mind, take another look at the theorem we started with.

Theorem [Division Algorithm]. Given any strictly positive integer d and any integer a, there exist unique integers q and r such that

$$a = qd + r,$$

and

$$0 \le r < d\,.$$

One might be excused for racking one's brains for quite some time to figure out what this theorem "really means." And yet it simply describes the process of dividing one integer a (the "dividend") by another d (the "divisor") to get a quotient (q) and a remainder (r). For instance,

$$\begin{array}{rr} & 36 \\ 29 & \overline{\big)\,1052} \\ & \underline{87} \\ & 182 \\ & \underline{174} \\ & 8 \end{array}$$

In this example, we have $a = 1052$ and $b = 29$. The theorem above states that there exist unique numbers q and r such that

$$1052 = 29q + r \quad \text{and} \quad 0 \le r < 29.$$

The calculation shows that in fact, $q = 36$ and $r = 8$.

The dividend a for the Division Algorithm is allowed to be negative. In this case, the quotient will also be negative (or possibly 0), but the remainder is still required to be positive.

$$\begin{array}{r|r} & -5 \\ \hline 29 & -121 \\ & -145 \\ \cline{2-2} & 24 \end{array}$$

This calculuation represents the equation $-121 = 29(-5) + 24$.

Operational, Procedural, and Notational Definitions

The statement of the division algorithm as given in the theorem describes very explicitly and formally what long division is. To borrow a word from physics, the description of long division by the two conditions $a = qd + r$ and $0 \leq r < d$ is **operational**. Given two numbers, for instance, 1052 and 29, the conditions give a very explicit way of testing whether or not 36 is the quotient and 8 the remainder when the first number is divided by the second.

On the other hand, these two conditions are not **procedural**: they do not provide a recipe for actually finding the quotient and remainder. In fact, until one recognizes what the theorem is describing as simply long division, it is not obvious that the theorem is even true. Good definitions in mathematics must always be operational (at least in some idealized sense), but they are often not procedural.

Analogously, consider the definition of the absolute value function give in many college algebra courses:

$$|x| = \begin{cases} x \text{ if } x \geq 0 \\ -x \text{ if } x < 0 \,. \end{cases}$$

This definition is both operational and procedural, since it describes exactly how to compute $|x|$. And yet most students find it completely bewildering. (Students who do make some attempt to make sense out of it often think it says, "If x is positive, leave it alone, and if it's negative, make it negative." They forget that if x is negative, then $-x$ if positive: for instance, $-(-13) = 13$.)

Unless I consider a student fairly bright, when he asks me what the above definition of the absolute value function really means, I usually say, "Just make x positive by throwing away the minus sign if it has one." One can say that this second definition ("throw away the minus sign") is operational and procedural, since it certainly enables one to easily compute the absolute value of a number. So what's the advantage of the first one, which seems so bewildering?

Well, for one thing, the first definition of the absolute value function is stated in terms of **intrinsic** properties of numbers, whereas the second definition is **notational:** it is stated in terms of the way we write numbers down. The second definition works fine when we want to computer the absolute value of a concrete number written down specifically, but it's not so useful when we want to talk about numbers in generality, or we have a number that's not described in concrete form. (For instance, if we say "Let x be the smallest solution of the equation $x^2 - x = 5$," there's no minus sign in the way we've named x, even though one can show that it is negative.) Most important, the first definition is useful for proving theorems, whereas the second is not.

Nonetheless, I think that giving the first definition in college algebra books is approaching mathematics in an overly formalistic way that drives many students away.

By the time a student gets to Number Theory, however, it's time he should begin to learn the kind of language that is used in contemporary mathematics.

In general, there are often two approaches to defining a mathematical entity. The procedural approach gives a method for actually calculating or constructing that entity. Alternatively, an entity can be defined by specifying a set of rules which characterize its "behavior" (as it were). **The mathematics of the Twentieth Century tends to put emphasis on the rules by which entities behave, rather on the way they are constructed.**

As an example of this modern approach, one might describe the natural logarithm as a function L with the properties that $L(xy) = L(x) + L(y)$ for all x and y, and $L'(1) = 1$. This definition says nothing about how to actually compute the natural logarithm. In fact, until one has proved the appropriate theorem, it is not even clear that this natural logarithm function even exists. However once the necessary existence theorem has been proved, this modern definition enables one to derive all the important formulas for the logarithm function (including the formulas for its derivative and integral) in a very clean manner.

The description of long division given in elementary school is notational. From a mathematician's point of view, it doesn't describe what long division really is in an intrinsic way, as the theorem does. Furthermore, the elementary school definition would not make any sense to someone who used Roman numerals or some other non-standard way of writing numbers. (Although the use of Roman or Chinese systems for representing numbers is not of great practical concern, there is non-decimal representation of numbers which is of quite great importance — namely the representation of numbers in computers. If one were required to write a computer program to do long division, one would not want to start from the sixth-grade procedure, although that procedure — suitably modified — might provide some useful insight.)

The description of the division algorithm by the conditions $a = qd + r$ and $0 \leq r < d$ is complete without reference to any illustrative example. It's also important to realize, though, that for us human beings, simple examples, such as the example of long division given above, are an important aid in understanding mathematics. As soon as one gives a simple numerical example for the Division Algorithm, the student or the mathematician will almost instantly say, "Aha!" The example appeals to a different sort of understanding, which is more natural for human beings. Unfortunately, in many advanced mathematical texts the reader has to construct such examples for himself. (One can be sure, however, that the author has constructed such examples. It's just that it's considered a little amateurish — perhaps insulting to the reader's intelligence — to include them in an advanced text.)

The Importance of the Division Algorthm

The Division Algorithm talks about the form of division one first learns about in elementary school, where one gets both a quotient and a remainder. Later on, after one learns about fractions, it may seem like a rather simple-minded way of thinking about division. I know it seemed like that to me. In particular, I was always annoyed by getting that damned remainder term. Somehow having a remainder made the division process imperfect, and that was inconsistent with my idea of mathematics as a subject where things are always exact.

In Number Theory, however, this is the way of looking at division which is most useful. And in fact that apparently annoying remainder turns out to be often much more important than the quotient. In computer science, and certain parts of mathematics, ones writes

$$a \bmod d$$

to denote the remainder when a is divided by d. For instance, $1052 \bmod 29 = 8$. Most books on number theory do not use this notation, but instead write $1052 \equiv 8 \pmod{29}$.

As a simple example of why this concept might be useful, consider the question, "If June 21, 1997 is a Saturday, then what day of the week will July 4 come on in the year 2000?" To answer this, first calculate, taking into account that the year 2000 will be a leap year, that there are $13 + 3 * 365 + 1 = 1109$ days between June 21, 1997 and July 4, 2001. Then divide 1109 by 7 and see that the remainder is 3, so that $1109 \bmod 7 = 3$. This tells us that there are a certain number of full weeks (158, in fact, since 158 is the quotient when 1109 is divided by 7) plus 3 more days from June 21, 1997 until July 4, 2001. The full weeks don't matter, so July 4, 2000 will be three days later than Saturday, namely Tuesday.

This example may seem of rather limited value, but similar situations arise in mathematical applications quite frequently, since many mathematical entities are cyclical in the same way as the days of the week are. For instance, if one were to ask what the 500^{th} digit in the decimal expansion of $1/13$ is, is would be foolish to compute all 500 digits. Since the decimal expansion of $1/13$, viz .076923076923..., has a cyclic pattern of length 6, one only need note that $500 \bmod 6 = 2$, so that the 500^{th} digit in this expansion will be the same as the second digit, namely 7.

Why We Need a Proof for the Division Algorithm

Although almost all books refer to the Division **Algorithm**, it is actually not the algorithm — the technique — which is being referred to. Instead, what is important is the theorem given above that says that a long division problem always has an answer.

Since this theorem simply describes a process that everyone learns to do in elementary school, one wonders why anyone would feel it necessary to prove it. Furthermore, at first it seems almost impossible that a theorem like this could be proved except by explaining the method for actually finding the quotient and remainder. I'm not sure I'd know how to approach it myself if I hadn't seen the proof so many times.

One rather unimaginative answer to the question of why we need to prove the Division Algorithm might be "Without a proof, you can't really know that the theorem is true." In my opinion, professors who give this answer are encouraging students' common belief that all mathematicians are a little bit crazy. I think that most students would say, "First of all, everybody knows that it's true and it's perfectly obvious. Secondly, the theorem has in fact been proved, and the proof is written down in reference books, and it's been taught in courses thousands of times, so why do *I* have to know the proof? Am I supposed to check the proof just to make sure that there's not some mistake that the rest of the world has missed?"

Mathematicians are usually not satisfied just to know that something is true, they want to know *why* it's true. Students are not mathematicians, however, and most of them don't even want to *become* mathematicians. So I don't know whether it's right or wrong for students to simply accept certain facts without proof. I think that probably for a lot of students, it's perfectly okay.

The reason I want to go through the proof of the Division Algorithm is not because I think that students are, or should be, skeptical, but because the proof illustrates some important ways of thinking.

The Proof of the Division Algorithm

Let's look at the theorem again and see why it looks so intimidating.

Theorem [Division Algorithm]. Given any strictly positive integer d and any integer a, there exist unique integers q and r such that $a = qd + r$ and $0 \leq r < d$.

One thing to notice right away is that, like many important theorems in mathematics, this makes a statement about **existence** and also one about **uniqueness**. So it's actually saying two things. It's typical for existence-uniqueness theorems that these two things have to be proved separately. Usually uniqueness is the more straightforward part to prove (the Fundamental Theorem of Arithmetic is an exception), whereas it's often rather bewildering to figure out how to prove that something exists.

Another thing to notice is that we are solving for **two** quantities: q and r. Now if we had two equations to satisfy, this would make sense. But there's really only one equation: $a = qd + r$, where a and d are given.

If we think about this situation as a problem in simultaneous equations, then we can remember that one approach is to find q first and then solve for r by substitution. In fact, once we know q then we get

$$r = a - qd.$$

There doesn't seem to be any good way of finding q, though. In fact, after a while one might notice that q can be chosen quite arbitrarily. One could choose $q = 0$ and $r = a$ or $q = 1$ and $r = a - d$. This is in fact typical in mathematics: when there are more unknowns than equations, that usually means that there are lots of different solutions.

When one looks at the example we started with, though, one sees that $q = 0$ and $q = 1$ are not correct values for the quotient when 1052 is divided by 29. The problem is that we have been considering only the equation

$$a = qd + r$$

instead of looking at the **whole** theorem. The values $q = 0$ and $q = 1$ don't work in the above example because they give the values 1052 and 1023 for r, and these don't satisfy the condition

$$0 \leq r < d.$$

It may seem what I've been doing in these comments is what I often do with my students in class, namely leading them up the garden path. But in fact, there's a very important insight to be learned from what we've just done. Namely, *in the Division Algorithm, having* ***two*** *unknowns q and r is a smokescreen. The Division Algorithm is actually a statement about only* ***one*** *variable q.*

Having the Division Algorithm stated in the usual way, with two variables q and r, is much more convenient in terms of applying it. But for purposes of figuring out how to prove it, it's much more enlightening to restate it in the following form:

Theorem [Division Algorithm]. Given any strictly positive integer d and any integer a, there exists a unique integer q such that $0 \leq a - qd < d$.

The equation $a = qd + r$ has now disappeared from the theorem, because we've used it to replace r by $a - qd$.

Notice that the conclusion of the theorem, $0 \leq a - qd < d$ actually consists of two statements: $a - qd \geq 0$ and $a - qd < d$. We are required to satisfy both these conditions simultaneously.

Remember again that this theorem makes an assertion about **existence** and one about **uniqueness**, and the existence statement seems the more bewildering.

To see why the two conditions of the Division Algorithm can be satisfied simultaneously, let's look at the usual procedure for computing q and r. In analyzing any system or process, it's often enlightening to notice what goes wrong when one makes a mistake. So what would happen in the example given above we mistakenly computed the quotient as 35?

```
      3 5
  29 |1 0 5 2
      8 7
     -----
      1 8 2
      1 4 5
     ------
        3 7
```

(In fact, in practice one often makes this sort of mistake when doing long division, especially if the divisor ends in a 9 or an 8.)

Why do we immediately recognize that this calculation is incorrect? *Because the remainder is too large.*

And what does the fact that the remainder is too large tell us? *That the quotient can be made bigger.*

We need to formalize this insight.

Lemma. If $a - qd \geq d$ for a certain value of q, then we can replace q by $q' = q + 1$ and still satisfy the condition $a - q'd \geq 0$.

If you're really following the logic here, then this lemma should seem like common sense. (If a certain value for the quotient in a division problem yields a remainder bigger than the divisor, this means that the quotient can be made larger without getting a negative remainder.) And its proof is just the application of that common sense.

Proof. If $a - qd \geq d$ and $q' = q + 1$, then $a - q'd = a - (q+1)d = a - qd - d \geq d - d = 0$.

Now let's attempt a rather unsophisticated proof of the existence part of the Division Algorithm.

Attempted Proof of the Division Algorithm (existence only). Keep making q larger until $a - qd < d$. If we take the first q that makes this true, then $a - qd \geq 0$.

As a criticism of this, we can note that it's fine to say "Keep making q larger," but where do we start? Well, we might start with $q = 0$. This won't work if the dividend a is negative, which is allowed, but for the time being suppose that a is positive. Okay, so what do we mean by "the first q"? That seems to be a nitpicking question, but in mathematics, nitpicking is often enlightening. What we mean by "the first q" is that q works (i. e. $a - qd < d$) but $q - 1$ doesn't (i. e. $a - (q-1)d \geq d$). **Okay, now how do we know that there is such a q?** That's the really killer question, but we can also ask how we know that if we choose q this way then $a - qd \geq 0$? Unfortunately, that doesn't quite come from the Lemma above.

Things will work a little better if we slightly restate the thinking, even though this restatement may seem a little less natural.

Second Attempted Proof. Choose the largest possible q such that $a - qd \geq 0$. Then $a - qd < d$ because of the above Lemma. (Suppose by way of contradiction that $a - qd \geq d$. Then the Lemma says that we can replace q by $q' = q + 1$ and $a - q'd \geq 0$. But this contradicts the assumption that q was the largest possible choice.) Therefore this q satisfies the conditions stated in the theorem.

At this point, there's only one piece of nitpicking left: *How do we know there is a largest possible q such that $a - qd \geq 0$?*

The reason is that by assumption, $d > 0$. (In the theorem, the dividend is allowed to be negative but the divisor is not.) So as q keeps getting larger, $a - qd$ keeps getting smaller and smaller. So eventually it has to become negative.

Is the previous paragraph rigorous, or does it need further proof? Well, it's actually okay. But there's a way of making it more formal. This is by using the following fundamental principle in Number Theory:

The Well Ordering Principle for Natural Numbers. Any non-empty set of positive (or non-negative) integers contains a smallest number.

Taking this for granted for the moment, we can write a formal proof for the existence part of the Division Algorithm.

Proof of the Division Algorithm (existence only). Consider the set of all numbers of the form $a - qd$, such that q is an integer and $a - qd \geq 0$.

There do exist numbers in this set: for instance, if a is positive then a is in the set (choose $q = 0$), and if a is negative then $a - ad = -a(d-1) = |a|(d-1)$ is in the set (choose $q = a$) since by assumption d is strictly positive and so $d - 1 \geq 0$.

Since we've seen that the set of integers of the form $a - qd$ such that $a - qd \geq 0$ is not empty, by the Well Ordering Principle, this set has a smallest number $a - qd$. Then by assumption $a - qd \geq 0$.

We claim that for this particular q, $a - qd < d$. In fact, if $a - qd \geq d$, then by the Lemma above we can replace q by $q' = q + 1$ and still have $a - q'd \geq 0$. But if $q' = q + 1$ then $a - q'd$ is smaller than $q - qd$ (because $q' > q$ and $d > 0$, so $a - q'd < q - qd$), so if $a - q'd \geq 0$ this would contradict the fact that we have already chosen the smallest possible non-negative number of the form $a - qd$.

This proves the claim that $a - qd < d$, and that proves the existence part of the Division Algorithm theorem. $\boxed{\checkmark}$

The proof of the division algorithm, which has been explained in such excruciating detail here, is actually simple common sense. One can state it informally as saying, "Subtract the highest possible multiple of the divisor from the dividend so that the resulting remainder is positive. Then this remainder is then necessarily smaller than the divisor, because if it weren't then one could subtract a still higher multiple of the divisor and still have a positive remainder."

This is a proof of the Division Algorithm that can be understood by elementary school students, if it's accompanied by a simple example or two. In a lot of ways, I think the paragraph above is a better proof than the formal one developed previously, because when one understands this proof the Division Algorithm really seems obvious — it really "makes sense." But the disadvantage of this two sentence proof is that in order to understand it, one has to really *think* about what it says. It is not explicit, it is not precise, and in some repects it is vague and can be interpreted ambiguously. And yet at the same time, in some ways it conveys more meaning than the formal proof.

Uniqueness. The uniqueness part of the Division Algorithm is already fairly clear from the comments so far. Namely, given a and d, there can be only one value of q such that $0 \leq a - qd < d$ since if one has such a q and then makes it any larger, then $a - qd < 0$, and if one makes it any smaller then $a - qd \geq d$. (One should carefully work through the inequalities that show this. The key thing to note is that since we're working with integers, if $q' > q$ then $q' \geq q + 1$, and consequently $q'd \geq qd + d$.)

It's more enlightening, though, to frame the uniqueness proof in a slightly different way. In general, whereas proofs of existence can be quite diverse, there is a standard format that proofs by uniqueness almost always follow. Namely, to prove that a certain entity (q, in this case) is unique, one assumes that one has two of them and then proves that these two have to be the same.

In this format, the proof of uniqueness in the Division Algorithm is as follows:

Proof of Uniqueness. For given a and d (with $d > 0$) there can exist only one pair of integers q and r such that

$$a = qd + r \quad \text{and} \quad 0 \leq r < d \,.$$

PROOF: Note first that since r is uniquely determined by q (since it is required that $r = a - qd$), what we really need to show is that there exists a unique value of q such that $0 \leq a - qd < d$. Now suppose that q and q' both satisfy this condition, i. e. $0 \leq a - qd < d$ and $0 \leq a - q'd < d$ as well. Then by subtraction we see that since $a - q'd \geq 0$,

$$(q' - q)d = (a - qd) - (a - q'd) \leq a - qd < d \,,$$

and likewise

$$(q'-q)d = (a-qd)-(a-q'd) \geq 0-(a-q'd) = -(a-q'd) > -d$$

since $a-q'd < d$.

Together, these two inequalities say that $(q'-q)d$ is an integer strictly between $-d$ and d. Since $d>0$, one can divide through by d to get

$$-1 < q'-q < 1\,.$$

Since $q'-q$ is an integer, this implies that $q'-q=0$, i. e. $q'=q$. This finishes the proof that q is unique, and as previously noted it follows automatically that r must also be unique. $\boxed{\checkmark}$

Another Point of View

Once an elementary school student learns about fractions, he learns to write down division in a new way. Intead of writing

$$\begin{array}{r}
36\\
29\,\overline{)\,1052}\\
\underline{87}\\
182\\
\underline{174}\\
8
\end{array}$$

he now writes

$$\frac{1052}{29} = 36\,\tfrac{8}{29}\,.$$

Once I learned this in elementary school, I immediately decided that this was the grown-up, sophisticated way of doing division, and that the quotient-remainder form was only for those who didn't know any better. It wasn't until I became a mathematician that I learned that both forms are useful.

In any case, we can now formally express the relationship between these ways of doing division. If we use $\lfloor x \rfloor$ to denote the "greatest integer" in x (i. e. the value of x rounded off to the next closest integer less than or equal to it), then we can see from the above example that if we divide an integer a by a divisor d to get a quotient q and remainder r, then

(1) $$q = \lfloor \frac{a}{d} \rfloor$$

(2) $$\frac{r}{d} = \text{ the fractional part of } \frac{a}{d}.$$

This should be clear from the example, however it is an interesting exercise to see how one would write a formal proof of this fact.

The key point here is that before one can have a proof, one must have formal definitions for all the concepts involved. Here, one needs to define the greatest integer function **operationally**.

For given x, $\lfloor x \rfloor$ is uniquely defined by the following conditions: (1) $\lfloor x \rfloor$ is an integer; (2) $\lfloor x \rfloor \leq x < \lfloor x \rfloor + 1$.

Now notice that if $a = qd + r$ and $0 \leq r < d$ then

$$q \text{ is an integer}$$

$$q \leq q + \frac{r}{d} = \frac{qd+r}{d} = \frac{a}{d}$$

$$\frac{a}{d} = \frac{qd+r}{d} < \frac{qd+d}{d} = q + 1\,.$$

Thus q satisfies the conditions that show that $q = \lfloor a/d \rfloor$. Therefore if q and r are defined according to the conditions in the Division Algorithm, then $q = \lfloor a/d \rfloor$ and it then follows immediately that $\frac{r}{d}$ is the fractional part of $\frac{a}{d}$. (This is not the slickest possible proof, but it shows the value of having concepts defined operationally.)

Further Comments on the Well Ordering Principle

Many students have little difficulty in accepting the fact that every (non-empty) set of positive integers has a smallest element. The trouble is, in fact, that some students have *too* little difficulty accepting it, and are equally willing to believe, for instance, that every set of positive integers has a largest element, or that every set of positive *real numbers* has a smallest element. Neither of these is true. For instance the set

$$1,\ 2,\ 3,\ 4,\ \ldots$$

of all positive integers does not have a largest element. And the set

$$1,\quad \frac{1}{2},\quad \frac{1}{3},\quad \frac{1}{4},\quad \frac{1}{5},\quad \ldots$$

of all real numbers of the form $1/n$, for n a positive integer, does not have a smallest element. (This set does not contain 0.)

What makes the Well Ordering Principle true is that although one can have an infinite increasing sequence of positive integers

$$1,\ 2,\ 3,\ 4,\ \ldots$$

and an infinite decreasing sequence of positive real numbers

$$1, \quad \frac{1}{2}, \quad \frac{1}{3}, \quad \frac{1}{4}, \ldots,$$

it's not possible to have an infinite sequence of positive integers that keeps getting smaller forever. (If a strictly decreasing sequence contains the value 500, for instance, then there are only 499 possible smaller values, so after at most 499 steps the sequence would have to either terminate or stop getting smaller.)

Another way of looking at it is to notice that obviously every *finite* set of numbers has both a largest and-ordered a smallest element. It's only with infinite sets that anything might possibly go wrong. But when one is looking for the smallest member of a set of positive integers, for practical purpose every set might as well be finite. This is because if the set one is concerned with contains 500, for instance, then in looking for the smallest member one might as well ignore all numbers which are larger than 500. This gives at most 500 possibilities to look at, and among these 500 (or fewer) numbers, there would have to be a smallest one. Notice that this reasoning only works, though, because we're only considering *integers*.

One can also give a proof of the Well Ordering Principle by mathematical induction, although it's a bit tricky.

Proof of the Well Ordering Principle by Mathematical Inducation. The proof is by contradiction. Suppose that S is a non-empty set of positive integers and assume by way of contradiction that S does **not** contain a smallest element. We will prove by "complete" induction that for all n, $n \notin S$, which is a contradiction to the assumption that S is non-empty.

(1) Basis Step: $1 \notin S$. Otherwise, clearly 1 would be the smallest element in S, and we're assuming that S has no smallest element.

(2) Induction Step: Now suppose that for a given positive integer n it is know that all the integers $1, 2, \ldots, n$ do not belong to S. Then $n+1$ can't belong to S either, otherwise $n+1$ would be the smallest element in S (since $1, \ldots, n$ are not in S) and we're assuming that S has no smallest element.

Therefore, by the principle of mathematical induction, no positive integer belongs to S. But this is a **contradiction** to the assumption that S is non-empty.

Since we have seen that the assumption that S has no smallest element leads to a contraction, it follows that S must contain a smallest element. $\boxed{\checkmark}$

The Well Ordering Principle is actually a variation on the principle of mathematical induction. Any proof using mathematical induction can easily be converted into one using the Well Ordering Principle. The following example will show the pattern:

Theorem. The sum of the first n odd integers equals n^2.

Proof by Induction. The theorem is true when $n = 1$ since the sum of the first 1 odd integer is just 1, and $1 = 1^2$.

Now assume that the theorem is true for a given value of n:

$$1 + \cdots + (2n-1) = n^2 .$$

($2n-1$ is the n^{th} odd number.) Then the sum of the first $n+1$ odd numbers will be

$$[1 + \cdots + (2n-1)] + (2n+1) = n^2 + (2n+1) = (n+1)^2 ,$$

showing that the theorem is also true for $n+1$.

Therefore, by the principle of mathematical induction, the theorem is true for all n. $\boxed{\checkmark}$

Proof by the Well Ordering Principle. We assume by way of contradiction that there exists a positive integer n such that the sum of the first n odd integers is not n^2. Then the set of such "bad" n is not empty, so by the Well Ordering Principle there is a smallest bad n. This first bad n can't be 1, since (the sum of) the first odd integer does in fact equal 1^2, so 1 is not bad. Thus if n is the first bad integer then $n > 1$, so $n-1$ is a positive integer. Since n is the smallest of all the bad integers, $n-1$ can't be bad. This means that the sum of the first $n-1$ odd integers does equal $(n-1)^2$. Then to get the sum of the first n odd integers, we add the n^{th} odd integer, namely $2n-1$, to the sum of the first $n-1$, which we have just seen is $(n-1)^2$, thus getting the total $(n-1)^2 + (2n-1)$. But

$$(n-1)^2 + (2n-1) = (n^2 - 2n + 1) + (2n-1) = n^2$$

showing that the sum of the first n odd integers is n^2 for this particular n, a **contradiction** to the assumption that this n is bad.

Since the assumption that the theorem is not true for some n leads to a contradiction, we see that the theorem must be true for all n. $\boxed{\checkmark}$

The essential core of these two proofs is identical. It is only the format that is different. For this particular theorem, the use of the Well Ordering Principle seems more awkward, since it requires a proof by contradiction.

The pattern of this proof, where one assumes that the desired theorem is false for at least one value of n and then gets a contradiction, is very typical for proofs using the Well Ordering Principle. As one sees more examples, this pattern will gradually seem less unnatural and one can be a little less wordy in spelling out all those details which are always the same for every theorem.