The purpose of this note is to explain the "long form" proof format. You will use this format for homeworks. The format has a particular structure, designed to make it easier to get the overall proof structure right, and to make it easier to pinpoint any errors.

Before your proof, state (as a theorem) the fact that you are going to prove. Make absolutely sure that the fact, as you write it, is well-defined — that is, what you write has a mathematically precise meaning.

Before you write a complete proof, you must already have in mind a line of reasoning that establishes that the fact is true. The line of reasoning must be complete (covering all possibilities) and without logical gaps. The goal of the written proof is to communicate your line of reasoning to the reader. The fundamental guiding principle is this: *from what you write, the reader should be able to easily follow your line of reasoning, and to easily verify that it is correct.*

(Of course, to figure out a line of reasoning, it is often useful to write partial proofs for yourself, just to guide your thinking.)

Write down your proof as a sequence of small steps. Number the steps for reference. Each step is either

– a definition, a statement of fact, a comment to guide the reader, or

– the first line of a *block* (such as an *assumption*; more on that later), or

– some combination of the above.

A *definition* gives a name to some quantity. Make sure the definition is mathematically precise, with no unwanted ambiguity. The definition then holds throughout the remainder of the block in which it occurs. (Just like a declaring a variable in a computer program makes it available for the remainder of the block in which the declaration occurs.)

For each *statement of fact*, what you write should be mathematically precise. Such a statement asserts that *the fact follows directly from the facts established in previous steps* (and is therefore true). "Directly" means that a competent reader, who has the intended background, should be able to easily see that the fact follows without having to think too hard. Usually, the statement of fact is accompanied by a justification to guide the reader.

We'll explain more about *blocks* later. Here is an example of a theorem with a long-form proof.

Theorem 1. *The number $x = \log_9 12$ is irrational. (That is, the x such that $9^x = 12$.)*

Proof (long form).

1. Assume for contradiction that x is rational. *block: proof by contradiction*
2. Then, by definition of "rational", there are integers p and q with $q > 0$ such that

$$\begin{aligned} p/q &= \log_9 12. \\ 9^{p/q} &= 12 && \textit{(raising 9 to the power of each side)} \\ 9^p &= 12^q && \textit{(raising each side to the power of q)} \end{aligned}$$

3. Because $q > 0$, the right-hand side 12^q above is an even integer.
4. But the left-hand side 9^p cannot be an even integer.
5. This is a contradiction. So our assumption in Line 1 must be wrong, so x is irrational. □

The proof consists of just one block. Step 1, an assumption, is the first line of the block. (This block is an example of *proof by contradiction.*) After the first step, each step of this proof is a statement of fact, which follows directly from the fact in the previous steps (as discussed on Page 1). The last step (Step 5) completes the proof of the theorem by establishing the fact that the theorem states (x is irrational). The proof assumes the reader is familiar with basic properties of integers, and uses such properties without explaining why they hold.

Often, the first step in proving a statement is to identify the logical structure of the statement. The logical structure of the statement will guide the structure of the proof. The statements we'll want to prove in this class are, generally, expressible in first-order logic, using just existential and universal quantifiers ($\exists$, $\forall$) and logical operators (and, or, not, if/implies, if and only if).

This brings us to the subject of *blocks* within the proof. The previous example used just a single outer block. But blocks can also nest. That is, blocks can have sub-blocks. In the next example, the main (outer) block of the proof has as a sub-block, consisting of steps 2.1–2.6. We can refer to it as Block 2.

Note how the steps within a sub-block are numbered by adding an additional level of numbering. This *always* indicates a nested block. And each block must be one of just *four* types, as explained next.

(Below "MCS" is the 2017 draft of *Mathematics for Computer Science*. It has a section on proofs.)

Theorem 2. *For all positive integers n, if $\log_7 n$ is rational, then it is an integer. (MCS Problem 1.7)*

Proof (long form).

1. Consider an arbitrary positive integer n. *block: $\forall$*

2.1. Assume that $\log_7 n$ is rational. *block: "if then"*

2.2. Let p and q be integers with $q > 0$ such that $p/q = \log_7 n$, so

$$\begin{aligned} p/q &= \log_7 n \\ 7^{p/q} &= n && \textit{(raising 7 to the power of each side)} \\ 7^p &= n^q && \textit{(raising each side to the power of q)} \qquad (1) \end{aligned}$$

2.3. Each integer has a unique prime factorization, so Equation (1) implies that n is some power of 7.

2.4. That is, $n = 7^i$ for some integer $i > 0$.

2.5. Substituting that into Equation 1 and expanding gives

$$\begin{aligned} 7^p &= (7^i)^q = 7^{iq} \\ p &= iq && \textit{(taking the } \log_7 \textit{ of each side)} \\ p/q &= i && \textit{(dividing each side by } q > 0\textit{)} \\ \log_7 n &= i && \textit{(using } \log_7 n = p/q\textit{).} \end{aligned}$$

2.6. It follows that $\log_7 n$ is an integer.

3. By Block 2, if $\log_7 n$ is rational, then $\log_7 n$ is an integer. □

The theorem statement has the logical structure "for all $n \in S$, $Q(n)$" where S is the set of positive integers, and $Q(n)$ is of the form "if $A(n)$ then $B(n)$", where $A(n)$ is "$\log_7 n$ is rational", and $B(n)$ is "$\log_7 n$ is integer." That is, the theorem statement is a "for all" statement, containing an "if then" statement.

The blocks in this proof are of two types. The main, outer block (Steps 1 through 3) proves the "for all". This block is a "for all" block. The inner block, Block 2 (Steps 2.1 through 2.6), proves the "if then" statement. Block 2 is an "if then" block.

"For all" blocks ($\forall$). Consider any statement of the form $\forall n \in S,\ Q(n)$, where S is some set and Q is some predicate.

Suppose we know that n is some element of S, but we know nothing else about n (that is, n is an *arbitrary* element of S). Suppose that, knowing only that n is an element of S, we can show that n has property $Q(n)$. Then our reasoning can be applied to *any* element of S, and it would show that Q holds for that element. So it must be that Q holds for all elements of S.

This gives us the following proof principle. Suppose in our proof we start a block with a step such as "Let n be an arbitrary element of S", and then within that block we prove that $Q(n)$ holds. Then, *outside* the block, we can conclude that "$\forall n \in S,\ Q(n)$" must be true. That is, all elements of S have property Q.

This is the proof principle used in the previous example. It is why, from the outer block of the proof, the theorem follows (for every positive integer n, if $\log_7 n$ is rational, then it is integer).

To let the reader know that we are starting a "for all" block, we always include the qualifier "arbitrary" in the first step. For example, "Let x be an **arbitrary** positive integer."

"If then" blocks. Suppose we are considering some given property A. Whether or not we know that A holds, we can explore what the logical consequences will be *if* A does indeed hold. To do this in a long-form proof, we start a new sub-block with a step of the form

"Assume A holds."

Then, throughout that sub-block, we can use A as if it is a known fact. Suppose that, *under that assumption*, within the sub-block, we are able to prove some statement B. Upon reflection, we can see that *if A does indeed hold*, then the reasoning in the block would show that B holds. So, if A does indeed hold, then B must also hold. That is, *outside the block* (no longer under the assumption), we can conclude "if A, then B".

This is the proof principle used in the inner block, Block 2, of the previous example. Specifically, just *after* Block 2, Step 3 used this principle, with Block 2, to conclude that *if* $\log_7 n$ *is rational*, then *it is integer*. Such a block is called an "if then" block.

"Proof by contradiction" blocks. A "proof by contradiction" block starts by assuming some property A, just like an "if then" block. Suppose that within the block (under the assumption that A holds) we can somehow prove a logical contradiction. Then we can conclude, outside the block, that A must be false. (A logical contradiction is a statement that must be false. Often, it takes the form "B and not B" for some proposition B.)

For an example, see the block used for the proof of Theorem 1 on page 1. There we assumed that x was rational, and under that assumption we proved a contradiction. We concluded that x is not rational.

To guide the reader, we always start the first step in the block with the phrase "Assume **for contradiction** that…".

Blocks for proof by cases. The fourth and final type of block is used for case analysis. Suppose you know that "C_1 or C_2" holds, and you want to show that some proposition X holds. You may not know which of C_1 or C_2 holds, but you know at least one has to.

First, consider the case that C_1 holds. Start a sub-block with the assumption that C_1 holds. Under that assumption (within that sub-block) show somehow that X holds. Then you know that, if C_1 holds, then X holds.

Next, consider the other case, that C_2 holds. Start a sub-block with the assumption that C_2 holds. Under that assumption (within that sub-block), show somehow that X holds. Then you know that, if C_2 holds, then X holds.

Finally, since you know that C_1 or C_2 holds, and you know that in either case X must hold, you can conclude (without any assumptions) that X holds.

This is the principle of proof by cases. It extends naturally to situations with more than two cases.

When you introduce a block for a proof by cases, always use a phrase such as "**Consider the case** that …" to let the reader know. If necessary, add additional steps with comments to help the reader understand the structure of the cases.

Next is an example, following MCS section 1.7. Here's the setup. We assume that, for pair of people in the world, either the two people have met, or they have not. For any given group G of people, if every pair in G has met, call G a *club*. If every pair in G has not met, call G a group of *strangers*.

Theorem 3. *Every group of six people includes either a club of three or a group of three strangers.*

Proof (long form).

1. Let G be an arbitrary group of six people. *block:* ∀
2. Say that G is *good* if F includes a club of 3 or a group of 3 strangers. Our goal is to prove G is good.
3. Fix x to be any one of the six people in G.
4. There two cases, each with two sub-cases:

5.1. Case 1: Suppose that, among the five other than x, at least three have met x. *block: cases*

5.2.1. Subcase 1.1: Suppose that no two among those three have met. *block: cases*

5.2.2. Then those three people are a group of strangers, so G is good.

5.3.1. Subcase 1.2: Otherwise some pair among those three have met. *block: cases*

5.3.2. That pair, and x, have all met each other, so form a club of three, so G is good.

6.1. Case 2 (Case 1 doesn't hold): Among the five other than x, at least three have *not* met x. *block: cases*

6.2.1. Subcase 2.1: Suppose that every pair among those three have met. *block: cases*

6.2.2. Those three are a club, so G is good.

6.3.1. Subcase 2.2: Otherwise some pair among those three have not met. *block: cases*

6.3.2. That pair, and x, have all not met, so form a group of three strangers, so G is good.

7. One of the Subcases 1.1, 1.2, 2.1, or 2.2 must hold, and G is good in each, so G is good. □

In doing proof by cases, if the reasoning for one case is *exactly* the same as a previously proved, symmetric case, you can simply point out that the reasoning is the same, instead of repeating it.

Later we will introduce lemmas within proofs to make proofs more modular (just as subroutines are used to make programs more modular).

Mind the gap! After you've written a proof, put it aside for a while, then come back with fresh eyes to check it. Put yourself in the place of the reader, who does not already know what you have in mind. The three main things to check are (i) that each statement and definition is mathematically precise, (ii) that the logic is correct and complete (the reasoning covers all cases), and (iii) that there are no large gaps (steps that don't follow directly from previous steps). Checking a proof carefully and skeptically is a learned skill. Here's a checklist that will help you avoid common errors:

1. *From what you've written, the reader will be able to easily reconstruct and verify your line of reasoning.*
2. Before the proof there is a theorem stating what is proved.
3. Each step in the proof is one of these types: a statement (with justification), a definition, a comment, or an assumption (starting a block). Or a combination thereof.
4. Each definition and statement is mathematically precise and well-defined, and each statement follows directly, by the justification given in that step, from facts already proved (either by domain-specific reasoning, or by one of the proof patterns from this proof guide).
5. The proof defines each name (variable) that it uses, before it uses it. It refers to a variable only within the block where the variable is defined (or in blocks within that block).
6. Each assumption is the first step of an appropriately nested block.
7. Each block is of one of these types: cases, if then, for all, or contradiction. (No blocks for definitions!)
8. Don't assume the reader already knows anything specific to your question. Do assume the reader knows basic facts about the basic structures you are reasoning with (e.g., numbers, trees, graphs, sets, algorithms). Adjust the degree of detail appropriately for the reader you anticipate.
9. Use standard mathematical structures: numbers, tuples, sets, sequences, permutations, graphs, trees, paths, matrices, etc. Avoid non-standard terminology, whose meaning won't be clear.
10. Use (precise!) English rather than formal notation, unless you can't say what you need to say precisely in words. It can be useful to define a new term for a concept that you use throughout your proof, but use definitions sparingly — it can be hard for the reader to keep track if there are too many.

Table 1: **Cheatsheet: the twelve universal proof principles.** For each logical operator (and, or, implies, not, for all, there exists), there are two proof patterns: one ("to show") that says how to prove a statement with that operator, the other ("to use") that says how to use such statement in a proof (to deduce other statements from that statement).

To show "A and B"

If you have proved both "A" and "B" (separately), then you can deduce "A and B".

To use "A and B"

If you have proved "A and B", then you can deduce "A", and you can deduce "B".

To show "A or B"

If you have proved "A", or you have proved "B", then you can deduce "A or B".

To use "A or B" (proof by cases)

1. …
2.1. **case 2.** Assume A.
2.2. … *within this scope, A holds* …
… *somehow prove C* …
2.3. C
3.1. **case 3.** Assume B.
3.2. … *within this scope, B holds* …
… *somehow prove C* …
3.3. C
4. C *(by "A or B", and cases 2 and 3)*

To use "if A then B"

If you have proved "if A then B", and you have proved "A", then you can deduce "B".

To show "if A then B"

1. …
2.1. **Assume** A.
2.2. … *within this scope, A holds* …
… *somehow prove B* …
2.3. B
3. If A, then B. *(by block 2)*

Alternatively, show the contrapositive:
"if not B, then not A".

To use "$\forall x \in S.\ P(x)$"

If you have proved "$\forall x \in S.\ P(x)$", and you know $y \in S$ for some particular y, you can deduce $P(y)$ for that y.

To show "$\forall x \in S.\ P(x)$"

1. …
2.1. Let x be an **arbitrary** element of S.
2.2. … *within this scope, x is defined* …
… *somehow prove P(x)* …
2.3. $P(x)$
3. $\forall x \in S.\ P(x)$. *(by block 2)*

To show "$\exists x \in S.\ P(x)$"

Prove that $P(y)$ holds for a particular $y \in S$.

To use "$\exists x \in S.\ P(x)$"

1. Let y be an element of S such that $P(y)$.
2. … *Now you can use y* …
… *and that $y \in S$ and $P(y)$* …

To use "not A"

Prove "A and not A" to reach a **contradiction**.

To show "not B"

1. …
2.1. Assume **for contradiction** that B holds.
2.2. … *within this scope, B holds* …
… *prove any contradiction* …
2.3. contradiction
3. Not B. *(by block 2)*

Basic logic: ("≡" means "is equivalent to")

"if A then B" ≡ "if not B then not A"
"if A then B" ≡ "B or not A"
"not (A and B)" ≡ "(not A) or (not B)"
"not (A or B)" ≡ "(not A) and (not B)"
"not (not A)" ≡ "A"
"not $\forall x \in S.\ P(x)$" ≡ "$\exists x \in S.$ not $P(x)$"
"not $\exists x \in S.\ P(x)$" ≡ "$\forall x \in S.$ not $P(x)$"

Blocks delimit the scopes of assumptions. The general principle is that *whenever we want to explore the consequences of some assumption, we begin a new subblock, with a step that makes that assumption.* The block delimits the scope of that assumption. Recall the block types: "if then", "proof by contradiction", "proof by cases", and "for all $x \in S$". For the first three, it is clear what the assumption is. But for the "for all" block, what underlying assumption necessitates a block? That S is not empty!