Validity Counterexamples Extended:
Chain Arguments and Validity

We have now developed our first, informal test of validity: when assessing an argument for validity, using just what's known about the actual world and our imaginations, we to try to think up a ***validity counterexample*** for that argument – a ***possible situation where all the premises of the argument are true, but the conclusion is false***. If we manage to think of a validity counterexample, we know that the argument is **invalid**.

But given the way we've defined "validity counterexample" – referring to "the premises" and "the conclusion" – that definition only applies to little 'one-step' arguments, which move directly from one or more premises to the main conclusion. That is: we've only defined "validity counterexamples" for **non-chain** arguments.

Chain arguments don't proceed directly from the original (top) premises to the main conclusion. Instead, the original premises yield a sub-conclusion, which can then turn around and (perhaps with other premises) yield another sub-conclusion, and so on – leading only in the end to the main conclusion. Since chain arguments insert a third element – sub-conclusions – between the top premises and the main conclusion, it's not clear what a validity counterexample for a chain argument would look like– or if there even could be such a thing. So it's not clear how we apply our informal test of validity to chain arguments.

But it's actually quite simple to extend our informal test of validity to chain arguments. To see how, recall that a chain argument gets its name from the fact that it has two or more argument ***links***, connected ("chained") together. So consider the following argument diagram.

```
   (1)==>(2)
   (3)==>(4)
(2)+(4)==>∴(5)
```

An argument fitting this diagram is made of *three* argument links: the link from (1) to (2); the link from (3) to (4); and the link from (2) and (4) together, to the main conclusion, (5).

Now, there's an old saying that *a chain only as strong as its weakest link*. And this turns out to apply to validity as well.

> **Chain Argument Validity Rule**: a *chain* of arguments is valid if (and only if) every link in the chain is valid. (In other words: a chain argument is valid if (and only if) it is made *entirely* out of valid links.)

Each of the *links* in a chain argument will be a non-chain argument – a little 'one-step' argument *whose validity we already know how to test* (by searching for validity counterexamples). Since we can test each link of chain argument for validity, we can use the *Chain Argument Validity Rule* to test the entire chain argument. For example, in the argument diagrammed above, we test each of the links – (1) to (2), (3) to (4), and (2) and (4) to (5) – for validity using our informal test (trying to imagine a validity counterexample for that 'link' argument). If we succeed in finding a validity counterexample for **even one** of these argument links, we know that the whole chain argument will be invalid as well.

Here's an example.[1]

> Either the butler or the judge committed the murder. But the butler was passionately in love with the victim, so the butler couldn't have committed the murder. Hence the judge must have committed the murder.

We begin by diagramming the chain argument. The first sentence is not a combo sentence, so the whole sentence receives one number.

> **(1) Either the butler or the judge committed the murder**. But the butler was passionately in love with the victim, so the butler couldn't have committed the murder. Hence the judge must have committed the murder.

[1] From Trudy Govier, *Asking the Right Questions*.

The second sentence *is* a combo sentence; the second part is marked as a conclusion by conclusion marker "so" (as well as conclusion-marking "couldn't have"), making the first part a premise in support of that conclusion. The two parts thus receive different numbers.

> (1) Either the butler or the judge committed the murder. But **(2) the butler was passionately in love with the victim,** so **(3) the butler** could**n't** have **committed the murder**. Hence the judge must have committed the murder.

The last sentence is not a combo sentence, so the whole sentence receives one number.

> (1) Either the butler or the judge committed the murder. But (2) the butler was passionately in love with the victim, so (3) the butler couldn't have committed the murder. Hence **(4) the judge** must have **committed the murder**.

The diagram so far looks like this.

```
   (1)
(2)==>(3)
   (4)
```

(4) comes in a likely place for the main conclusion, and is marked by conclusion marker "hence" (as well as the conclusion-marking modal "must have"). So we take (4) as the main conclusion.

```
   (1)
(2)==>(3)
  ∴(4)
```

Since (1) and (3) currently look useless, the *No Useless Sentences* Principle says they must be supporting the main conclusion.

(2)==>(3)
(1)+(3)==>∴(4)

So the argument has two links (marked in the diagram by arrows). And the *Chain Argument Validity Rule* says: if either of these links is invalid, the whole chain argument is invalid.

We thus ask, of each link, whether we can find a validity counterexample for that little argument. Very likely you will not find a counterexample for the second link.

1. Either the butler or the judge committed the murder.
3. The butler [did]n't commit the murder.

————————————

∴ 4. The judge committed the murder.

Intuitively, this little argument looks perfectly **valid**: if the premises *were* true, the conclusion would have to be true as well.

By contrast, it's easy to think of counterexamples for the first argument link.

2. The butler was passionately in love with the victim.

————————————

∴ 3. The butler [did]n't commit the murder.

Even if the premise was true, the conclusion could still be false. For example, perhaps the butler's love for the victim was unrequited – a situation so intolerable that it led the butler to murder. Or perhaps the butler loved the victim but discovered that the victim was secretly involved with someone else, leading the butler to commit murder in a jealous rage.

It seems obvious enough that being in love won't in itself rule out the butler as the murderer. And having found a counterexample showing this little argument is **invalid**, the *Chain Argument Validity Rule* says that the whole chain argument inherits this invalidity.

Summary: Chain Arguments and Validity

- **Chain Argument Validity Rule**: a *chain* of arguments is valid if (and only if) every link in the chain is valid.
- **Testing a Chain Argument for Validity**: apply the informal test of validity (trying to imagine a validity counterexample) to each link in the chain argument. If there is a validity counterexample for *even one* link in the chain, the whole chain argument is invalid.