# Accumulate/write running total for a variable

## What it does:

The purpose of this function is to accumulate a running total for a variable and then output the total. It should be used as follows:

1) Identify a delayed function (i.e. @d/x/) and point it to this function.

2) You may also identify a corresponding delayed function (i.e. @d/y/) and point it to the associated count/xx function.

3) Each occurrence of the delayed function subtracts 1 from the total count. Each occurrence of the count/xx function adds 1 to the total count. For example, to count x 3 times, you must include the @d/x/ 3 times in the template.

4) When you want to print the total, include a variable and point it to this function (i.e. @v/totalx/).

## Used in template:

```
@d/xx/
@v/totalxx/
```

| | | |
|---|---|---|
| TOTALXX | @V/ | FF COUNT/01 |
| XX | @D/ | FF SUBTRACT/01 |

## What the user sees:

The user will normally select something with a number, for example mitigating factors that would reduce the total. By using the function subtract/01 and the count/01 we come up with a -1. Each @d/xx/ variable counts as 1. If I wanted 3, I would put @d/xx/@d/xx/@d/xx/. One set for each number. The @d/ variable can be anything you want, it just has to be unique.

## Output:

```
-1
```

### FF SUBTRACT/00 – 30

You can keep track of up to 30 calculations at the same time. The Count and the Subtract must be the same number. FF COUNT/02 and FF SUBTRACT/02.