Task 13: reinventing assert

Create A C macro that behaves just like `assert` from `<assert.h>`:

- `assert(1 == 2);` ⇒ `test.c:3: main: Assertion '1 == 2'failed.` is printed on stderr, the program aborts
- `assert(1);` ⇒ no output, everything is fine

Some Tips:

- The special macros `__FILE__`, `__LINE__` and `__FUNCTION__` can give you information about the current location in the source code
- You can use `abort();` (`#include <stdlib.h>`) to abort the program
- You can use `#macro_param` to turn the macro parameter into a string

Some More Tipps

- Remember that we can print on stderr using `fprintf(stderr, "foo: %s", "bar");` (`#include <stdio.h>`)
- We can use the `do { } while(false)` trick to make the macro safer