ERRATA: Christopher W. Fraser and David R. Hanson, 'A Machine-Independent Linker', *Software—Practice & Experience*, **12**, 4 (Apr. 1982), 351–366.

The implementation of `link` described on pp. 363–365 contains an error. In order for relocation to be correct, the sum of the `len` commands for each segment that appear in the temporary files after pass one must be equal in *all* of the temporary files. This invariant is not maintained when a *new* segment is introduced in pass 1 after `len` commands for other segments have been emitted into the existing temporary files. The result is incorrect relocation for inter-segment references in the new segment. This error can be illustrated by linking the files `a.o` and `b.o`, given in the paper, with `c.o`:

```
.seg bss
data
.len bss 1
```

If `c.o` is linked last, the reference to `data` will not be relocated properly because the temporary file for the `bss` segment does not include `len` commands to reflect earlier contributions to the `data` segment.

This error occurs infrequently because in most applications of `link`, e.g., compiler output, the first input file mentions all of the segments in `seg` commands before any `len` commands are emitted. This usage is equivalent to creating all of the temporary files before reading any of the input.

The error can be fixed by emitting the `len` commands necessary to maintain the invariant when a temporary file is created. The pseudo-code

```
for each segment id1 do
        output '.len id1 base_id1' to id's temporary file
```

should be inserted immediately following lines 8 and 20 in Figure 1.