

```
1: // $Id: scanfbug.c,v 1.9 2013-09-25 14:54:29-07 - - $
2: //
3: // NAME
4: //     scanfbug - illustrate bug in scanf.
5: //
6: // DESCRIPTION
7: //     Uses %lg to read and print numbers or errors out for non-numbers.
8: //     But fails to recognize a lone + or - on input.
9:
10:
11: #include <assert.h>
12: #include <stdio.h>
13: #include <stdlib.h>
14:
15: int main (void) {
16:     int exit_status = EXIT_SUCCESS;
17:     for (;;) {
18:         double number;
19:         int scanrc = scanf ("%lg", &number);
20:         if (scanrc == EOF) break;
21:         if (scanrc == 1) {
22:             printf ("number = %.15g\n", number);
23:         }else {
24:             exit_status = EXIT_FAILURE;
25:             char buffer[1024];
26:             scanrc = scanf ("%1023s", buffer);
27:             assert (scanrc == 1);
28:             printf ("bad input \"%s\"\n", buffer);
29:         }
30:     }
31:     return exit_status;
32: }
33:
34: /*
35: //TEST// (echo 34 92.6 123foob + - 38 - + +3 -33 infinity \
36: //TEST// +infinity -infinity nan - 43 end \
37: //TEST// | ./scanfbug; echo status = $?) >scanfbug.lis 2>&1
38: //TEST// mkpspdf scanfbug.ps scanfbug.c* scanfbug.lis
39: */
40:
```

[illegible]

```
1: + echo 34 92.6 123foob + - 38 - + +3 -33 infinity +infinity -infinity na
n - 43 end
2: + ./scanfbug
3: number = 34
4: number = 92.6
5: number = 123
6: bad input "foob"
7: bad input "-"
8: number = 38
9: bad input "+"
10: number = 3
11: number = -33
12: number = inf
13: number = inf
14: number = -inf
15: number = nan
16: bad input "43"
17: bad input "end"
18: + echo status = 1
19: status = 1
```