

What is Wrong with the Code?

```
if ((num - 1) > i && 0 != u[i]) {
    printf (".");
}
```

What's wrong?

The programmer **confused the region index *i* with the index among non-zero regions.**

All but the last non-zero region should be followed by a gap, but *i* also counts zero regions.

How to Fix the Bug

To fix the bug quickly, we can

- **add a separate variable `non_zero`** to index non-zero regions,
- **initialize `non_zero` to 0** when *i* is set to 0,
- **increment `non_zero` only when we see a non-zero region**, and
- **compare `non_zero` to `(num - 1)`** to decide whether to print a gap.

Fixing the Bug

Here's how it might look
(except for declaration and initialization).

```
if (0 != u[i] &&
    (num - 1) > non_zero++) {
    printf (".");
}
```

With this change, the code passes
all 6,391 of my tests as well.

Fixing the Bug

Alternatively,

```
◦ compress zero regions out at the start,
◦ making the false equivalence true.
for (i = 0; 4 > i; i++) {
    if (0 != u[i]) {
        u[num++] = u[i];
    }
}
for (i = num; 4 > i; i++) {
    u[i] = 0;
}
```