

[sign up](#) [log in](#) [tour](#) [help](#)

Stack Overflow is a community of 4.7 million programmers, just like you, helping each other.

Join them; it only takes a minute:

[Sign up](#)

Join the Stack Overflow community to:

Ask  
programming  
questions

Answer and help  
your peers

Get recognized for your  
expertise

## C - How to format strings using printf() to get equal length in the output?

I have two functions, one which produces messages like `Starting initialization...` and another which checks return codes and outputs `"Ok"`, `"Warning"` or `"Error"`. However, the output that is produced is of the different length:

```
Starting initialization...Ok.  
Checking init scripts...Ok.
```

How can I get something like this:

```
Starting initialization...      Ok.  
Checking init scripts...      Ok.
```

`c` `string` `printf` `pretty-print`

edited Oct 26 '15 at 17:09



demonplus

2,636 7 20 35

asked Nov 27 '09 at 15:38



psihodelia

8,819 16 72 131

### 6 Answers

You can specify width on string fields, e.g.

```
printf("%-20s", "initialization...");
```

and then whatever's printed with that field will be blank-padded to the width you indicate.

The `-` left-justifies your text in that field.

answered Nov 27 '09 at 15:42



Carl Smotricz

44.3k 11 82 138

1 The example input strings are longer than 20 characters.. – [PP](#) Nov 27 '09 at 15:50

61 Alas, I'm not clever enough to count that high. However, I trust the asker to be intelligent enough to extrapolate from my example :) – [Carl Smotricz](#) Nov 27 '09 at 16:03

`printf` allows formatting with width specifiers. e.g.

```
printf( "%-30s %s\n", "Starting initialization...", "Ok." );
```

You would use a negative width specifier to indicate left-justification because the default is to use right-justification.

answered Nov 27 '09 at 15:43



PP.

7,751 5 29 52

There's also the `%n` modifier which can help in certain circumstances. It returns the column on which the string was so far. Example: you want to write several rows that are within the width of the first row like a table.

```
int width1, width2;
int values[6][2];
printf("|%s%n|s%n|\n", header1, &width1, header2, &width2);

for(i=0; i<6; i++)
    printf("|%*d|%*d|\n", width1, values[i][0], width2, values[i][1]);
```

will print 2 columns of the same width of whatever length the two strings `header1` and `header2` may have. I don't know if all implementations have the `%n` but Solaris and Linux do.

edited Mar 26 '15 at 8:41

answered Nov 28 '09 at 19:31



Patrick Schlüter

7,090 22 31

Additionally, if you want the flexibility of choosing the `width`, you can choose between one of the following two formats:

```
int width = 30;
//no truncation
printf( "%-*s %s\n", width, "Starting initialization...", "Ok." );
//truncated to the specified width
printf( "%-.*s %s\n", width, "Starting initialization...", "Ok." );
```

edited Oct 26 '15 at 17:16

answered Mar 5 '15 at 4:49



lifebalance

658 5 24

There's also the rather low-tech solution of counting adding spaces by hand to make your messages line up. Nothing prevents you from including a few trailing spaces in your message strings.

answered Nov 27 '09 at 15:44



Carl Smotricz

44.3k 11 82 138

Start with the use of Tabs, the `\t` character modifier. It will advance to a fixed location (columns, terminal lingo). However, it doesn't help if there are differences of more than the column width (4 characters, if I recall correctly).

To fix that, write your "OK/NOK" stuff using fixed number of Tabs (5? 6?, try it), then return (`\r`) without new-lining, and write your message.

edited Nov 27 '09 at 15:50

answered Nov 27 '09 at 15:43



jpinto3912

1,061 2 9 17

4 There's always the danger that a string is longer than a tab stop thereby forcing the resulting tab and text to be mis-aligned. – PP Nov 27 '09 at 15:45

it doesn't work ! – psihodelia Nov 27 '09 at 15:46

Not really... first write the tabbed OK/NOK, then `/r`, now write the phrase, now `/n`. The phrase could overwrite the OK/NOK. But your way is way better, I didn't know about negative justification. – jpinto3912 Nov 27 '09 at 15:48

@Carl, thanks, I keep doing this mistake... – jpinto3912 Nov 27 '09 at 15:49

---

@psihodelia, yep, 'cause I had / iso \... my bad. – [jpinto3912](#) Nov 27 '09 at 15:51

---

**protected** by [Yu Hao](#) Sep 23 '13 at 14:26

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 [reputation](#) on this site.

Would you like to answer one of these [unanswered questions](#) instead?