O'REILLY®

Errata

Errata for *21st Century C*

Submit your own errata for this product.

Print

Confirmed Errata | Unconfirmed Errata

The errata list is a list of errors and their corrections that were found after the product was released. If the error was corrected in a later version or reprint the date of the correction will be displayed in the column titled "Date Corrected".

The following errata were submitted by our customers and approved as valid errors by the author or editor.

Color key: Serious technical mistake Minor technical mistake Language or formatting error Typo Question Note Update

Version	Location	Description	Submitted By	Date submitted	Date corrected
PDF	Page xiv 3rd paragraph	"As of this writing, most compilers support C99 plus or minus a few caveats; the long double type seems to cause a lot of trouble, for example." The long double type is not a new feature in C99. It was introduced by the 1989 ANSI C standard, if not earlier. The only implementation I know of that has trouble with long double is MinGW, because it combines the gcc compiler with the Microsoft runtime library. Both handle long double consistently, but they give it different sizes, making the combination inconsistent. **Note from the Author or Editor:** This should've read "long long double". Fixed by BK in commit 98f8f38.	Keith Thompson	Nov 22, 2014	May 15, 2015
Mobi	Page location 3154/13446 (22%) A command line above last paragraph in	To delete the side branch one doesn't use "git delete other_branch", but "git branch -d other_branch" or it's stronger version "git branch -D other_branch" ("git branch -d --force other_branch" in latest git version) if the other_branch was not merged into current one. There is no "git delete" command.	Jakub Narębski	Mar 14, 2015	May 15, 2015

	Their Branches" section of Chapter 4. Version Control	about what to do after merging, and git gives increasingly friendly advice about when "git branch -D" is necessary, so I don't think it necessary to add to the text.			
Mobi, Other Digital Version	Page location 10009/13446 (73%) Example 11-23. The group_s object. (groups.w), paragraph starting with "@ I played around a lot with different rules,..."	"The standard distance is the L_2 norm, aka Euclidian distance, meaning that the distance between (x_1, y_1) and (x_2, y_2) is $\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$. This is L_3, $\sqrt[3]{(x_1-x_2)^3+(y_1-y_2)^3}$." The definition of L_3 norm in two dimensions lacks absolute value sign, namely to be a norm the equation for it needs to take absolute value of difference between components, namely $\sqrt[3]{\|x_1-x_2\|^3 + \|y_1-y_2\|^3}$ This does not matter for L_2 (Euclidian) distance because (x_1-x_2)^2 is positive, and symmetrical change 1 <-> 2. This doesn't matter for code, because it uses Apophenia library: apop_vector_distance(g->position, position, .metric='L', .norm=3). **Note from the Author or Editor:** Good catch! Fixed by BK in version 73219f0.	Jakub Narębski	Mar 21, 2015	May 15, 2015
Safari Books Online	IV In the "Some Logistics" section of the preface of the 2nd edition.	This is minor, but since you mention "pedantic readers"... Shouldn't "misconstrued incorrectly" simply be "misconstrued"? In what way could a reader misconstrue something correctly? :-) -sk **Note from the Author or Editor:** Brilliant. This is my favorite erratum so far, if only because it is my first meta-erratum. I suppose some people could be bad at misconstruing things, but I changed the text to just plain "misconstrued" as recommended. Fixed in revision a4254f7.	sk	Sep 17, 2016	
Safari Books Online	Ch13 Section: "The GNU Scientific Library"	"...writing down a distance function, wrapping it and all the relevant *metatdata* into..." metadata is spelled incorrectly. **Note from the Author or Editor:** Good catch. Fixed in revision d3511e9.	Yung-Jin (Joey) Hu	Apr 16, 2017	
Safari Books Online	Ch12 Section: "Atoms" Code listing: "Example 12-6"	In example 12-6 we are importing "openmp_getmax.c" from example 12-5 with the intent of using the function defined in "openmp_getmax.c" called `get_max()`. However, in code listing 12-6, `get_max_factors()` is used instead. I think in Example 12-6 we should be calling `get_max()` instead of `get_max_factors()`. **Note from the Author or Editor:** This is correct---in fact, get_max_factors doesn't even exist in that code segment. The sample code gets this	Yung-Jin (Joey) Hu	Apr 16, 2017	

Safari Books Online	Ch 1 Ch 1, Section "Using Makefiles", the "Your Turn" box.	Ack! No, no, no. The printf statement in K&R, 1st edition is: printf("hello, world\n"); No caps. No punctuation besides a comma. (At least you didn't use an exclamation point.) Please. ;-) **Note from the Author or Editor:** Indeed, I mis-transcribed from memory. Fixed by author in revision 5fb8fd8, also in chapter 7.	Charles Sharp	Oct 01, 2018	
Safari Books Online	1 Ch 1, Section "Using Makefiles", first "Usage:" bullet	In the bullet, the text states: " If you are using GNU Make, you have the option of capitalizing the name to Makefile if you feel that doing so will help it to stand out from the other files." BSD make, Solaris make, and GNU make all have search orders. All three will use a lower-case named makefile before using the upper-case version (in the absence of a '-f' filename). We used to always use an upper-case named Makefile. That way, if a developer needed to run a couple of experiments, they could just copy the Makefile to makefile, make a few changes, try it out, and then delete it. This is guaranteed behavior stated in the man page. **Note from the Author or Editor:** Now that you point it out, I re-checked, and capital-M Makefile is fully POSIX-standard. Thanks. Fixed by the author in revision 3e62046.	Charles Sharp	Oct 01, 2018	
Safari Books Online	1 Example 1-1	Linking with the math library (libm) for the referenced example doesn't appear to be necessary. I.e.: gcc erf.c -o erf -g -Wall -O3 -std=gnu11 Compiles with no errors or warnings. However, if the argument to 'sqrt' is set to an uninitialized variable, e.g.: int value; .....sqrt(value).... the compilation invocation *does* generate an error. ('undefined reference'). Adding the lm flag corrects the problem. It appears the compiler is optimizing beyond what one might expect. gcc version: gcc (Ubuntu 7.3.0-27ubuntu1~18.04) 7.3.0 **Note from the Author or Editor:** Wow, this is pretty cool, and I'm going to check what else gcc is willing to assume. I slightly reworded the text from 'you have to' to 'you can', but this is non-standard so I'm reluctant to discuss it further. (e.g., my local version of clang still requires -lm). Change made in commit a4f1183.	Mark Kohalmy	Apr 25, 2019	
PDF	Page 7 command line at the bottom: ./configure --host=ming32	is command line: ./configure --host=ming32 should be: ./configure --host=mingw32 **Note from the Author or Editor:** Fixed by BK in commit fba5fc9	Anonymous	Nov 20, 2014	May 15, 2015
PDF	Page 17	in the section where he talk about makefile variables or	karim reefat	Jan 17, 2020	

		by make will know new environment variables but won't know any makefile variables. " i think this is not right , the command line makefile variables will be visible to any child programs called by make as any new environment variables , let take an example: if i call make from the command line like this: make CFLAGS="-g -Wall" and inside the makefile : $(P): echo $(CFLAGS) the result will be displaying "-g -Wall" on the screen . so the value of CFLAGS is visible to echo even it's set on the command line when calling make which make it makefile variable and here echo will be the child program . so i think that this exception is wrong . **Note from the Author or Editor:** You make a good point. Checking the POSIX standard now, it says that "Before the makefile(s) are read, all of the make utility command line macro definitions (except the MAKEFLAGS macro or the SHELL macro) shall be added to the environment of make." Thank you for pointing out this exception. ---- Fixed by BK in revision 33cab4e.			
PDF	Page 18 Last paragraph before the "The Rules"-section	The description of $< is incorrect. $< is the first prerequisite for the rule in question. It doesn't matter if that file has been modified or not. $? on the other hand are *ALL* the prerequisites that are newer than the target. See e.g. https://www.gnu.org/software/make/manual/html_node/Variables.html This should demonstrate the difference: $ cat makefile foo: bar baz qux @echo First prerequisite: $< @echo Newer than target: $? @touch $@ $ touch bar baz qux $ make First prerequisite: bar Newer than target: bar baz qux ("foo" is now up to date, but touch one or more of the files and the rule has to be executed again.) $ touch qux $ make First prerequisite: bar Newer than target: qux **Note from the Author or Editor:** Interestingly enough, GNU make and the POSIX spec differ here. My text is a paraphrase of POSIX [http://pubs.opengroup.org/onlinepubs/009695399/utilit]: "In an inference rule, the $< macro shall evaluate to the filename whose existence allowed the inference rule to be chosen for the target." The commenter is correct that GNU Make treats $< differently. From its manual: "The $< variable evaluates to the first prerequisite in the first rule for this target." Version 7c47753 puts the GNU Make definition in the text and the POSIX definition in a footnote (because I've asked around and can't find anybody using a non-GNU version of make).	Povl Ole Haarlev Olsen	Feb 14, 2016	
PDF	Page 18 3rd	the line : " gram name, darn it—but the oddness aside, you can see that it took little setup within " the	karim reefat	Jan 27, 2020	

		Note from the Author or Editor: Sentence rewritten by BK in revision 6c1da1e.			
PDF	Page 20 1st (real) paragraph and the preceding/follo single lines.	You forgot to mention the "-c" option, without which cc/c99/gcc will try to make an executable, not an object file. Luckily, make doesn't forget to add the "-c" option, if you ask it to build an .o file: $ cat makefile P=program_name OBJECTS= CFLAGS = -g -Wall -O3 LDLIBS= CC=c99 $(P): $(OBJECTS) $ touch program_name.c $ make -n program_name.o c99 -g -Wall -O3 -c -o program_name.o program_name.c **Note from the Author or Editor:** Thanks for catching this. Version cd4346b replaces the erroneous recipe with the POSIX recipe, $(CC) $(CFLAGS) -c $< and fixed the verbiage accordingly.	Povl Ole Haarlev Olsen	Feb 14, 2016	
PDF	Page 25 Last sentence before section "Using Libraries from Source"	There is a suggestion to set LDLIBS=-L/usr/local/lib -Wl,R/usr/local/lib. My ld command claims the -R is an unknown option. Page 16 (my PDF) shows how to use the -R flag with the LDADD variable. Is this what you really meant on page 25? It worked for me when I changed LDLIBS to LDADD, and then also used the LDLIBS= -L/usr/local/lib -lgsl -lgslcblas -lm. **Note from the Author or Editor:** Yes, LDADD is for make; LDLIBS is for the plain compiler. Fixed by BK in revision d45da72.	Paul Glezen	Apr 11, 2016	
Printed	Page 44 top line (undisp 3)	"Stop the display the of display item 3." Need to remove the second "the". **Note from the Author or Editor:** Another good catch. Fixed in version 76a2320.	Rick Ward	Nov 30, 2016	
Printed	Page 55 Note 3	"The GLib test harness provides some extra assertion macros, like the string comparison macro, g_assert_compstr, used here." macro should be g_assert_cmpstr **Note from the Author or Editor:** Yes, I wished for more vowels than there are. Fixed by BK in revision 4dab317.	Rick Ward	Nov 30, 2016	
PDF	Page 83 Figure 3-1 summarizes the story as a flow diagram.	In "Figure 3-1 summarizes the story as a flow diagram." the flow diagram starts with "Automake.am". I suppose this should be "Makefile.am" as the text talks about Makefile.am? **Note from the Author or Editor:** Yes, Automake.am is at this point archaic, and the autotools seem to have standardized on Makefile.am.	Christoffer Holmstedt	Jun 16, 2015	

Printed	Page 110 Example 5-1	I was not able to compile the code in dynamic.c. This was due to the use of the strcmp() used on line 20 and <string.h> not being included in the header. **Note from the Author or Editor:** Good catch: readline.h includes the strcmp declaration somewhere in there, but it should be explicit. fixed in book repository commit c0879ed and example repository commit c651032.	Artagan Malsagov	Nov 12, 2020	
PDF	Page 124 Part 2 introduction, paragraph concerning chapter 12	"The secret is in parallel threads, and Chapter 12 covers covers three systems for turning [...]" should be "The secret is in parallel threads, and Chapter 12 covers three systems for turning [...]" Typo: "covers" written twice. **Note from the Author or Editor:** Good catch! Fixed by BK in revision 0ce1e9b.	Christoffer Holmstedt	Jun 16, 2015	
Printed	Page 127 2nd paragraph	"When a program runs across this declaration in your code: int *a_pointer; the program will only do one of the above steps: * declare that an_array is a pointer" ====== I think it should say "declare that a_pointer is a pointer". **Note from the Author or Editor:** Good catch. Fixed in commit 7c7b040517e4.	Anonymous	Oct 23, 2014	May 15, 2015
Printed	Page 129 Bottom paragraph	** ignore my previous errata, which contained an error ** "Python generally copies scalars but aliases lists (unless you use copy or deepcopy." This is a bit pedantic, but strictly speaking Python aliases scalars: i = 1 j = i assert i is j It's true that incrementing i won't affect the value of j, so that they behave as if j were a copy, but that's because scalars are immutable, meaning that the increment operation changes where the reference points rather than changing the value of the referenced object: old_id = id(i) i += 1 assert i == 2 assert j == 1 assert id(i) != old_id assert id(j) == old_id **Note from the Author or Editor:** BK added a (hopefully) clarifying footnote in commit 2427e7d.	Ben Plommer	Nov 17, 2014	May 15, 2015
ePub	Page 185 discussion following directions NORTH SPUTH EAST WEST enum	The sections weighing pros/cons of enums vs strings mistakenly assume that 8-bit instructions would execute faster than 16-bit instructions. While a narrower bit width could in theory be executed faster, in practice modern computer architectures tend to schedule instructions under a uniform clock rate. So manipulating a 1, 2, 4, 8, 16, 32, or 64 bit structure on modern computers will all use the same clock time. This happens even with variable length instructions: Chip makers have learned to convert	Andrew Pennebaker	Nov 25, 2016	

		context of arrays of 8-bit vs 16-bit numbers. But in general, enums are considered an extremely efficient and safe tool for modelling categorical things. **Note from the Author or Editor:** Yes, it was an overstatement to claim that an 8-bit comparison is always faster than a 16-bit, and reworded the sentence accordingly: "So even if the speed argument were relevant, comparing chars would be as fast or faster than comparing enums." But I am leaving this sentence in place, because some of the books this chapter is responding to do push the efficiency of enums as a selling point for their use (sorry, no time to pull up references right now). I felt a sentence responding to this literature was worth including. Fixed in revision f845bd6.			
Printed	Page 187 Example 9-3, first comment	In the first comment of Example 9-3 in the 2nd Edition, First Release, the comment beginning "/* The declaration, to put into a .h file..." runs past the right-hand margin and to the edge of the page. Some of the comment seems to be missing. **Note from the Author or Editor:** Yeah, not sure what happened here. BK added a line break to fix this in revision ea0954f7.	Kevin Zembower	Mar 20, 2015	May 15, 2015
Printed	Page 218 2nd paragraph (Warning)	s/unit/uint/ "Formally, the C standard only reserves int..._t and unit..._t" should read "Formally, the C standard only reserves int..._t and uint..._t" -- Guy Shaw **Note from the Author or Editor:** Good catch! Fixed by BK in revision b316af8.	Guy Shaw	Apr 04, 2015	May 15, 2015
Printed	Page 219 2nd to last paragraph	"typdeffed" in first sentence presumably should be "typedefed", as in second sentence. **Note from the Author or Editor:** OK, Fixed by BK in commit 46841f2.	Ben Plommer	Dec 12, 2014	May 15, 2015
Printed	Page 224 footnote 3	Footnote leaves off in the middle of a sentence. "[...] compiler doesn't support it, via" **Note from the Author or Editor:** Oops, I had a in the wrong place. Fixed by BK in revision 10d7e2b. The remainder of the footnote, btw: .. then have the C preprocessor define <code>__attribute__</code> to be blank should Autoconf find the user's compiler doesn't support it, via #include "config.h" #ifndef HAVE__ATTRIBUTE__ #define __attribute__(...) #endif It goes on the declaration line of a variable, struct, or function, so if your function isn't declared before use, you'll need to do so.	Matt Steadman	Jun 23, 2015	

	(PDF), Footnote 23 (epub) - last sentence	_attribute_ to be blank should Autoconf find the user's compiler doesn't support it, via" **Note from the Author or Editor:** Oops, I had a in the wrong place. Fixed by BK in revision 10d7e2b. The remainder of the footnote, btw: .. then have the C preprocessor define <code>_attribute_</code> to be blank should Autoconf find the user's compiler doesn't support it, via #include "config.h" #ifndef HAVE_ATTRIBUTE_ #define _attribute_(...) #endif It goes on the declaration line of a variable, struct, or function, so if your function isn't declared before use, you'll need to do so.			
Printed	Page 237 1st paragraph	"In this example to to recurse through a directory..." should be "In this example to recurse through a directory..." duplicate "to". **Note from the Author or Editor:** Good catch, thanks. fixed in revision 881820b.	Anonymous	Jul 02, 2017	
ePub	Page 334 5th paragraph in the chapter 'Unicode Libraries'	The line "Recall that 8 bytes is not nearly enough to express all characters in one unit, so a single character is between one and six units long." should probably read "Recall that 8 bits is not nearly enough to express all characters in one unit, so a single character is between one and six units long." **Note from the Author or Editor:** Yes, good catch. Fixed by BK in commit fdabfbf.	Mario Landgraf	Nov 05, 2014	May 15, 2015
PDF	Page 336 token	I think that it's just small typo. There is no such function as "strtok_n". The C Standard defined strtok and POSIX strtok_r. My sugeestion is to rewrite this sentence as: "into tokens; strtok and strtok_r are designed for this" **Note from the Author or Editor:** On p 193, I discuss the various variants: "The POSIX standard provides strtok_r , and the C11 standard provides strtok_s ." The glossary should have mirrored this, mentioning strtok_r and strtok_s. Fixed by BK in commit 3b44e7c.	Grzegorz Szpetkowski	Dec 20, 2014	May 15, 2015
Printed	Page 339 Line 26	nyt_feed.c, Line 26 of page 339 (also line 45 of the listing downloaded from the github repository): char *rss_url = "h t t p : // rss.nytimes.com/services/xml/rss/nyt/HomePage.xml"; should read char *rss_url = " h t t p s : // rss.nytimes.com/services/xml/rss/nyt/HomePage.xml"; [sorry about the extra spaces, but the errata system will not allow URL's!!] In recent years it has become fashionable to only serve up the encrypted https flavour of your web pages. Requests for the insecure http flavour	bl4krat	Sep 04, 2020	

		Note from the Author or Editor: Good point about updating. Fixed in the github repository thanks to a bug report there. Fixed here (along with another stale NYT link) in revision c6bc38e.			
Printed	Page 345 Appendix A - C101 - The Structure	Further to Charles Sharpe's errata dated 1 Oct 2018: K&R's program to say 'hello' still only uses lower case and a comma, so you may want to correct the first listing in the C-primer too. Change: <code>printf("Hello, world.\n");</code> To: <code>printf("hello, world\n");</code> **Note from the Author or Editor:** OK, fixed by author in commit d582d40.	bl4krat	May 13, 2020	
PDF	Page 351 3rd paragraph	"Some things are surprisingly easy thanks to the array nature of strings. Given char* str="Hello"; you can turn a Hello into Hell by inserting a NUL character: str[4]='\0';" This is violation of 6.4.5/7 subclause (String literals) according to C11 Standard (referring to N1570 draft) as well as by previous C99 and C89 Standards. The result is undefined behaviour: "If the program attempts to modify such an array, the behavior is undefined." The valid way would be to replace string literal with string array like: char str[]="Hello"; // the number of elements is guessed by initializer. **Note from the Author or Editor:** Good point. Fixed by BK in repository, commit 4950db5.	Grzegorz Szpetkowski	Dec 17, 2014	May 15, 2015
Safari Books Online	351 Code segment on the bottom of the page	strncpy(str2, 100, str1); strncat(str2, 100, str1); should be: strncpy(str2, str1, 100); strncat(str2, str1, 100); **Note from the Author or Editor:** Good catch! Fixed by BK in revision 4cc7ce	Anonymous	May 10, 2015	May 15, 2015
PDF	Page 353 Appendix A: C 101 - page 353 - 8th paragraph	in the paragraph: You saw another example several times above: in the evaluation of printf("hello\n"), the expression is replaced by a zero on success, but the evalua- tion is useful for the side effect of changing the state of the screen. the part when he said: the expression is replaced by a zero on success, i think that expression printf("hello\n") will be replaced by the number 6 not zero because printf On success it will return the total number of characters not just zero. **Note from the Author or Editor:** Good catch. Fixed in revision dcdaffe.	karim reefat	Aug 20, 2019	