

# Errata

**Last updated 09/07/2015.**

Despite our best efforts to create a book with zero defects, our vigilant readers have pointed out some bugs. Please report any new errata to [Randy Bryant and Dave O'Hallaron](#). Note that some of these errors have been corrected in more recent printings.

We maintain errata for the English-language editions. Errata for editions in other languages can be found as follows:

- [Chinese](#), maintained by [Prof. Yili Gong](#) of [Wuhan University](#).

## North American Edition (ISBN 0-13-610804-0)

### • Preface

### • Chapter 1: A Tour of Computer Systems

- p. 26, Bibliographic notes, line 3. The second author of the Silberschatz operating systems book is Galvin, not Gavin.

*Posted 09/29/2013. Kaustabh Roy*

### • Chapter 2: Representing and Manipulating Information

- p. 52, Practice Problem 2.11, Part B: It should read "Why does this call to function `inplace_swap` set the array element to 0?"

*Posted 04/24/2011. Kanak Kshetri*

- p. 63, fourth line: It should read "... there is no guarantee of portability."

*Posted 11/04/2011. Ronald Jones*

- p. 82, second line. The book states that Niels Abel was a Danish mathematician. In fact, he was Norwegian.

*Posted 03/02/2011. Dag Henning Liodden Sørbo*

- p. 84, Figure 2.23. The last sentence of the caption should read "When it is greater than  $2^{w-1} - 1$ , there is positive overflow."

*Posted 02/01/2011. Dominique Smulkowska*

- p. 84, third line from bottom. Should read: "When  $x+y \geq 8$ , the addition has a *positive* overflow ..."

*Posted 06/17/2010. Yili Gong*

- p. 100, Section 2.4.1, paragraphs one (two lines after formula for decimal representation) and two (two lines after formula for binary representation): "weighted by positive powers of ..." should be "weighted by nonnegative powers of ...".

*Posted 08/03/2011. Dong Han*

- p. 100, Section 2.4.1, paragraph two, second line. The second-to-the-last binary digit in the sequence should be labeled  $b_{-n+1}$ , rather than  $b_{-n-1}$ .

*Posted 08/03/2011. Dong Han*

- p. 101, Figure 2.30. Label  $b_{-n-1}$  should be  $b_{-n+1}$

*Posted 08/03/2011. Dong Han*

- p. 114, second paragraph, second line. " $x+a \geq x+b$ " should be written " $x+^fa \geq x+^fb$ ".

*Posted 12/14/2012. Ashwin Nanjappa*

- p. 114, fourth paragraph, second line. "Other than NaN:" should be "Other than NaN and INF: "

*Posted 9/17/2013. LU Junlin*

- p. 120, Bit-level coding rules. The item "Casting, either explicit or implicit" should be moved from the list of forbidden operations to the list of allowed operations.

*Posted 05/24/2011. Yu Tanaka*

- p. 123, Problem 2.68. The example when  $n = 6$  should yield 0x3F.

*Posted 05/24/2011. Taketo Yoshida*

- p. 143. Solution to Problem 2.32. The function will give an incorrect results for all values of  $x$  when  $y = TMin$ , not just when  $x < 0$ .  
*Posted 01/04/2015. Oswaldo Olivo*
- p. 147. Solution to Problem 2.41. The first case should be when  $m > 0$ , and the second case is when  $m = 0$ .  
*Posted 06/20/2014. Libo Chen*
- p. 150, Solution to Problem 2.51D Should read: " $0.086 \times 2000 \approx 171$  meters".  
*Posted 06/17/2010. Yili Gong*

• **Chapter 3: Machine-Level Representation of Programs**

- p. 167, Figure 3.1, Size of "long long int" should be 8 instead of 4.  
*Posted 09/10/2012. David O'Hallaron*
- p. 175, Text above Figure 3.6. The last line should state "Instruction 1 reads parameter  $x_p$  from memory and stores it in register ..."  
*Posted 07/27/2011. Ji Bin*
- p. 176, Practice Problem 3.4. The last sentence of the problem statement should state "... the operation should change the size first ...."  
*Posted 04/11/2010. David Ramsey*
- p. 184, first line. It should read: "A more conventional method of setting up the *dividend* makes use of the `cltd` instruction."  
*Posted 03/28/2011. Yili Gong*
- p. 188, third paragraph, line 7. The parenthetical remarks are reversed. The first case is one of negative overflow, and the second case is one of positive overflow.  
*Posted 06/10/2010. Jonas Pfenninger*
- p. 200, Problem 3.20, part C: The reference should be to Figure 3.14(c).  
*Posted 10/13/2011. Ji Bin*
- p. 209, Aside. The equation at the middle of the bottom line should read  $T_{MP} = 2(T_{ran} - T_{OK})$ .  
*Posted 04/23/2010. David Ramsey*
- p. 210, abstract code at bottom of page. Instead of "*true-expr*," it should be "*then-expr*." Also, the label "done" should have the same indentation as the label "false"  
*Posted 11/18/2010. Lei Ma*
- p. 216, Last sentence of first paragraph. Should read "We recommend that you study the C procedure `switch_eg_impl` and how it relates to the assembly code version."  
*Posted 04/26/2010. Gordon Kindlmann*
- p. 222, Figure 3.22(b). The arrow should point to the box containing the value 0x800483e1.  
*Posted 09/27/2010. David O'Hallaron*
- p. 234, 6th example in table. The type of expression  $*(E+i-3)$  should be `int`, not `int *`  
*Posted 11/14/2010. Karl Pichotta*
- p. 243, paragraph starting with "To generate a pointer to an object ...." Last sentence should read: "For pointer  $r$  in register `%edx` and integer variable  $i$  in register `%eax`, we can generate ...."  
*Posted 04/23/2010. David Ramsey*
- p. 253, Code line written "`(int) (*fp)(int, int *)`";. The first set of parentheses should be omitted, yielding line: "`int (*fp)(int, int *)`";.  
*Posted 04/03/2012. Ronald Greenberg*
- p. 259, Practice Problem 3.43, line 6. Should state that register `%esi` is equal to 0x2, and register `%edi` is equal to 0x3.  
*Posted 11/01/2010. Mustafa Kazdagli*
- p. 272, IA32 code for `simple_l`. line 5. Comment should read "Retrieve  $y$ ".  
*Posted 04/26/2010. Gordon Kindlmann*
- p. 277, Last sentence of paragraph following heading "Arithmetic Instructions": "32-values" should be "32-bit values"  
*Posted 11/04/2011. Ronald Jones*
- p. 284, Practice Problem 3.50. The problem should read "Determine ALL valid function prototypes for `incrprob` ...."  
*Posted 06/12/2012. Qiangqiang Luo*
- p. 286, First paragraph, lines 6 and 10. In both places, "`call_proc`" should be "`proc`"  
*Posted 11/04/2011. Ronald Jones*
- p. 300, Problem 3.60. Clarification: the `imull` instruction on line 4 uses a three-operand format that is not documented in the text. It multiplies the first two arguments and stores the result in the third. This format can only be used when the first argument is a constant.

*Posted 04/15/2015. Guilherme Avelino*

- p. 301, Problem 3.62. Argument definition `Marray_t` `A` should be `int A[M][M]`

*Posted 03/06/2010. Randal E. Bryant*

- p. 311, Answer to Problem 3.12. It is also possible for `num_t` to be of type `long long`. For this case, the C standard dictates that the multiplication operates as if `x` is first converted to `long long`, and so the multiplication is performed based on signed arithmetic. Interestingly though, the bit-level representation of the result is the same as for the unsigned case. To see this, recognize in part B that when splitting a signed 64-bit value `y` into 32-bit values `y_h` and `y_l`, we should consider `y_h` to be a signed value, while `y_l` should be considered an unsigned value. Therefore, we want to compute the 64-bit product `x * y_l` with unsigned multiplication, while the low-order 32-bits of `x * y_h` is the same for both unsigned and signed arithmetic.

*Posted 07/05/2012. Qiangqiang Luo*

- pp. 312-313, Answer to Problem 3.14. Part A: when `data_t` is unsigned, the comparison could also be `>`. Part D: The comparison could also be `!=` for either `short` or unsigned `short`.

*Posted 10/17/2014. Gregoire Paradis*

- p. 315, Answer to Problem 3.19. We can see, in fact, that the computation of `13!` has overflowed, since `1,932,053,504 / 13 = 161,004,458.667`

*Posted 10/19/2011. Richard Callahan. Corrected version posted 04/25/2012. Yili Gong*

- p. 321, Answer to Problem 3.29. The last sentence of the first paragraph should read "That leaves only label `.L5` to match case `E`."

*Posted 05/08/2013. Rami Ammari*

- p. 322, Answer to Problem 3.32. There are eight different functions declarations that yield the same code: the return type can be either `int` or unsigned, argument `p` can be either `int *` or unsigned `*`, and argument `x` can be either `int` or unsigned. These choices can be made independently.

*Posted 07/31/2012. Qiangqiang Luo*

- p. 326, Answer to Problem 3.41. The listing of field names in the answer should be "`i c d j`".

*Posted 10/31/2010. David Selvaraj*

- p. 327, Answer to Problem 3.43. The value labeled "saved `%edi`" should be `00 00 00 02`. The value labeled "saved `%esi`" should be `00 00 00 03`.

*Posted 10/11/2018. Alexander Mentis*

- p. 330, Answer to Problem 3.50. Variable `q` can also be of type unsigned `long`, and hence there are eight valid function prototypes. The additional ones are:

```
void incrprob_s_ul(int x, unsigned long *q, int *t);
void incrprob_u_ul(unsigned x, unsigned long *q, int *t);
void incrprob_sl_ul(long x, unsigned long *q, int *t);
void incrprob_ul_ul(unsigned long x, unsigned long *q, int *t);
```

*Posted 06/12/2012. Qiangqiang Luo*

- p. 331, Answer to Problem 3.51A. The stack frame should not include a box for offset 0. This stack location holds the return address, which is considered part of the caller's stack frame. Here is a [PDF](#) rendition of the correct figure.

*Posted 06/16/2012. Qiangqiang Luo*

#### • Chapter 4: Processor Architecture

- p. 346, Sentence starting on 3rd line from bottom. Should read "Lines 15 to 24 show ...."

*Posted 06/11/2010. Yili Gong*

- p. 370 Aside tracing the `pushl` instruction. The second instruction byte should be `0x2f`, instead of `0x28`. The value of `rB` in the Fetch stage should be `0xf` instead of `0x8`.

*Posted 12/08/2010. Karl Pichotta*

- p. 373, Practice Problem 4.15, first sentence. Should read "We can see ... that the `rmmovl` instruction is the unconditional version ...."

*Posted 05/20/2010. Randal E. Bryant*

- p. 384, Figure 4.26. The values for constants `INOP` and `IHALT` are reversed. They should be 1 and 0, respectively.

*Posted 01/17/2012. John Ye*

- p. 391, Second paragraph of "Surveying SEQ". Line 6 should state "... the ALU must increment the stack pointer ...".

*Posted 08/19/2013. Yili Gong*

- p. 400, Figure 4.39, part (b). Label "`Cnd`" should be "`pCnd`".

*Posted 05/25/2012. Yili Gong*

- p. 415, Figure 4.49 The blue box under the heading "Cycle 6" should be labeled "W", not "M".  
*Posted 04/05/2011. Yang Firo*
- p. 418, fourth and fifth lines from bottom. The sentence should state "The value for register %ebx is also forwarded from the memory to the *decode* stage."  
*Posted 08/09/2015. Liemin Ma*
- p. 420, Figure 4.53. Error arrow should point to valB line.  
*Posted 05/25/2012. Yili Gong*
- p. 425, Figure 4.55. The label "M\_Bch" should instead be "M\_Cnd".  
*Posted 01/19/2015. Jiwen He*
- p. 441, Figure 4.68. The block labeled "stat" and the generated signal m\_stat should provide input to pipeline register W, not M. Here is a [PDF](#) rendition of the correct figure.  
*Posted 06/20/2014. Sankar Krishnan.*
- p. 452, Problem 4.46. The reference should be to the code written for Problem 4.45.  
*Posted 06/21/2010. Yili Gong*
- p. 461, Problem 4.9 Solution. The figure should have an OR gate, rather than an AND gate. Here is a [PDF](#) rendition of the correct figure.  
*Posted 07/27/2011. Pascal Garcia*
- p. 462, Solution to Problem 4.11. The second instruction byte should be 0xf4, instead of 0x84. The value of rA in the Fetch stage should be 0xf instead of 0x8.  
*Posted 12/15/2010. Dorukhan Gülöz.*
- p. 462, Solution to Problem 4.12. The second instruction byte should be 0xf0, instead of 0x08. The value of rB in the Fetch stage should be 0xf instead of 0x8.  
*Posted 12/15/2010. Randal E. Bryant.*
- p. 466, Solution to Problem 4.26. All four uses of "GOPS" should be replaced with "GIPS".  
*Posted 06/21/2010. Yili Gong*
- p. 466, Solution to Problem 4.27. The formula should use lower-case *k* throughout.  
*Posted 11/04/2011. Ronald Jones*
- p. 468, Solution to Problem 4.31, Last sentence should state: "... %eax will be set to the incremented stack pointer rather than the value read from memory."  
*Posted 08/11/2014. Sebastian Luy*
- p. 468, Solution to Problem 4.33, Second to last sentence should state: "... the conditional move source value 0x123 gets forwarded into ALU input valA, while input valB correctly gets operand value 0x321."  
*Posted 07/21/2015. Liemin Ma*

## • Chapter 5: Optimizing Program Performance

- p. 479, Second block of code in aside on "Optimizing function calls by inline substitution". Line four should read  
counter = counter + 4  
*Posted 05/10/2010. David Ramsey*
- p. 481, Figure 5.1. Comment on line 19 should read "For even n, finish remaining element."  
*Posted 06/13/2011. Zhengshan Yan*
- p. 503, assembly code for combine4 and its associated description. Variable "limit" should be "length", and the text should state that the value is held in register %rbp.  
*Posted 1/20/2011 and 10/13/2011. Karl Pichotta and Ji Bin*
- p. 504, First paragraph. Line 8 should read "... register %rbp is only used by the cmp operation ..."  
*Posted 05/10/2010. David Ramsey*
- p. 504, Under category **Read-Only**. Last sentence should read "The read-only registers for the loop in combine4 are %rax and %rbp."  
*Posted 05/10/2010. David Ramsey*
- p. 505, first line. Should state: "... the cmp operation and used by the **jb** operation ..."  
*Posted 07/09/2010. Yili Gong*
- p. 505, second full paragraph, third line. Should state: "... (cmp) and branch (**jb**) operations ..."  
*Posted 07/09/2010. Yili Gong*
- p. 505, second full paragraph, line 7. Should read: "... not taken."  
*Posted 11/04/2011. Ronald Jones*
- p. 506, line 12 of text. Sentence should read "The chain on the **right** would require a ..."  
*Posted 07/09/2010. David Ramsey*
- p. 511, Figure 5.17 legend. Symbols for "int \*" and "int +" are interchanged.  
*Posted 1/20/2011. Karl Pichotta*

- p. 515, Figure 5.22 legend. Symbols for "int \*" and "int +" are interchanged.  
*Posted 1/20/2011. Randal E. Bryant*
- p. 520, Figure 5.27 legend. Symbols for "int \*" and "int +" are interchanged.  
*Posted 1/20/2011. Randal E. Bryant*
- p. 523, Problem 5.8. The code computes the product of an array of  $n$  doubles, not integers.  
*Posted 11/13/2011. Ji Bin*
- p. 528, second line from bottom. Should state "... it increases the CPE by at most 3 clock cycles."  
*Posted 07/09/2010. Yili Gong*
- p. 533, Figure 5.32. The loop limit check on line 11 should be  $i < n$ .  
*Posted 10/13/2011. Ji Bin*
- p. 537, Figure 5.37. In Example A, there should be no lines between the boxes labeled "s\_data" and those labeled "load". Here is a corrected version ([pdf](#)).  
*Posted 02/10/2010. Randal E. Bryant*
- p. 543, Figure 5.38 caption. "resultss" should be "results".  
*Posted 02/18/2010. David O'Hallaron*
- p. 547, First paragraph, line 10. The formula should be  $177.57/4.72 = 37.6$  rather than  $173.05/4.72 = 37.6$ .  
*Posted 09/07/2015. Liemin Ma*
- p. 553, Solution to Problem 5.4A. "... in the manner of variable x in combine4" should be "... in the manner of variable acc in combine4"  
*Posted 05/25/2012. Yili Gong*
- p. 553, Solution to Problem 5.4A, code for combine3w. Line 8 of the code should contain the statement `*dest = IDENT;` to ensure that the value is stored even when the vector length is less than or equal to zero.  
*Posted 01/03/2013. Dominik Helm*

#### • Chapter 6: The Memory Hierarchy

- p. 574, second paragraph, "when the head arrives at the target sector" should be "when the head arrives at the target track".  
*Posted 12/19/2013. Lucas Bärenfänger*
- p. 585, second full paragraph, line 4. "increasing" should be "decreasing".  
*Posted 11/4/2011. Ronald Jones*
- p. 602, paragraph 2 (after the table). "The next three columns" should be "The next four columns".  
*Posted 07/23/2010. Yili Gong*

#### • Chapter 7: Linking

- p. 659, Aside: The acronym .bss stands for "Block Started by Symbol," not "Block Storage Start."  
*Posted 11/13/2014. Paul C. Anagnostopoulos*
- p. 660, last paragraph. Actually, the compiler assigns variables that are initialized to zero to .bss rather than .data. So the instance of x defined in function f() is stored in .bss, and not .data.  
*Posted 11/20/2012. Waiki Wright and Scott Wright*
- p. 670, paragraph 3, "referenced by main.o" should be "referenced by main2.o".  
*Posted 11/08/2011. Ji Bin and Yili Gong*

#### • Chapter 8: Exceptional Control Flow

- p. 709. Paragraph 3. The phrase "The handler maps the appropriate page of physical memory on disk into a page of virtual memory" should read "The handler maps the appropriate page of virtual memory on disk into a page of physical memory"  
*Posted 02/23/2012. jcw2001@163.com*
- p. 711. Figure 8.11, line 14. The comment should read "System call number 1" instead of "System call number 0".  
*Posted 05/24/2011. Michael Victor Zink*
- p. 721. Second to last paragraph. Process graphs are mentioned in passing without a proper discussion. Please see [Web Aside ECF:GRAPHS](#) on the [Web Aside page](#) for a more thorough treatment.  
*Posted 04/14/2011. Gordon Kindlmann*
- p. 726. Practice Problem 8.3. To enable all of the feasible outputs, you must follow each printf with an fflush(stdout).  
*Posted 12/18/2014. Len Hamey*
- p. 740. Last paragraph. The discussion of SIGINT and SIGTSTP is incorrect. Typing ctrl-c causes the kernel to send a SIGINT signal to each member of the foreground process group (not to the shell). Similarly for ctrl-z and SIGTSTP.

*Posted 09/29/2014. Godmar Back*

- pp. 747--751, Figures 8.31--8.33. The programs call a function (`printf`) that is not async-signal-safe inside a signal handler. This introduces a potential deadlock. Please see [Web Aside ECF:SAFETY](#) on the [Web Aside page](#) for details.

*Posted 09/01/2010. Godmar Back*

- p. 761, Program in Figure 8.39. According to the Single Unix specification, the statement `rc = setjmp(buf)` in line 14 is not standards-compliant because of the assignment to a local variable. It should be replaced with a switch statement `switch(setjmp(buf))`.

*Posted 07/29/2014. Yan Zhu*

- p. 767. Homework Problem 8.18. To enable all of the feasible outputs, you must follow each `printf` with an `fflush(stdout)`.

*Posted 12/18/2014. Len Hamey*

- p. 768. Homework Problem 8.21. To enable all of the feasible outputs, you must follow each `printf` with an `fflush(stdout)`.

*Posted 12/18/2014. Len Hamey*

- p. 769, Problem 8.22. "system returns the value 8" should be "mysystem returns the value 8".

*Posted 07/28/2010. Yili Gong*

- p. 771, Problem 8.26. The proposed handling of `ctrl-c` and `ctrl-z` is a simplification of the way that real shells work. With real shells, the kernel responds to `ctrl-c` (`ctrl-z`) by sending `SIGINT` (`SIGTSTP`) directly to each process in the terminal foreground process group. The shell manages the membership of this group using the `tcsetpgrp` function, and manages the attributes of the terminal using the `tcsetattr` function, both of which are outside the scope of this book.

*Posted 09/29/2014. Godmar Back*

- p. 772, Solution to Practice Problem 8.3. The sequence `bcac` is not possible. Strike the second to last sentence. The last sentence should be "There are three possible sequences: `acbc`, `abcc`, and `bacc`." Please see [Web Aside ECF:GRAPHS](#) on the [Web Aside page](#) for an example of the process graph.

*Posted 5/03/2010. David Ramsey and Gordon Kindlmann*

- p. 772, Solution to Practice Problem 8.4. Strike the sentence beginning "In general, any topological sort..." and the (incorrect) graph that follows. Please see [Web Aside ECF:GRAPHS](#) on the [Web Aside page](#) for an example of the correct process graph.

*Posted 5/03/2010. David Ramsey and Gordon Kindlmann*

## • Chapter 9: Virtual Memory

- p. 789, Bulleted list: "Step 3: the MMU" should be "Step 4: the MMU", and "Step 4: the cache" should be "Step 5: the cache".

*Posted 04/6/2010. David O'Hallaron*

- p. 790, two lines above Practice Problem 9.3: "in Figure 9.13(b)" should be "in Figure 9.13(a)"

*Posted 07/30/2012. Yili Gong*

- p. 800, line 5. L3 is actually 16-way set associative.

*Posted 07/30/2012. Yili Gong*

- p. 804, Fig 9.26. The start address for the 64-bit read-only Program text segment should be `0x00400000`.

*Posted 04/18/2011. Firo Yang and Yili Gong*

- p. 808, Fig 9.29(b). For clarity, the location of the mapped VM area for Process 2 should be clearly different than for Process 1.

*Posted 11/28/2011. Tao Du*

- p. 809, Fig 9.30(a). The location of the mapped VM area for Process 2 should match that of Part (b) and be consistent with Fig. 9.29(b).

*Posted 11/28/2011. Tao Du*

- p. 812, line 6. First argument to `mmap` should be `NULL` rather than `-1`.

*Posted 08/5/2014. Sankar Krishnan Shanmugam*

- p. 816, Fig 9.34 caption. "Padded regions of allocated blocks are shaded with stripes" should be "Padded regions of allocated blocks are shaded with a darker blue".

*Posted 07/3/2010. Yili Gong*

- p. 819, line 8 and line 2 in the aside. "over the first  $k$  requests" should be "over the first  $k+1$  requests"

*Posted 04/9/2012. Tao Du*

- p. 819, 3rd line from the bottom. "If the request in Figure 9.34(e) were for six words" should "If the request in Figure 9.34(e) were for eight words".

*Posted 07/30/2012. Yili Gong*

- p. 838, line 3, should be "continue coalescing with the free buddies."  
*Posted 04/6/2010. David O'Hallaron*
- p. 853, Solution to Practice Problem 9.1, second to last line, last column, should be " $2^{48} - 1 = 256T - 1$ ."  
*Posted 12/11/2012. Yili Gong*
- p. 856, Solution to Practice Problem 9.8, a right brace is missing from the last line.  
*Posted 12/15/2012. Rami Ammari*

- **Chapter 10: System-Level I/O**

- p. 872, Figure 10.7, `rio_readlineb`: Edge case when input line is `maxlen` bytes and not terminated by a newline. [Updated version of csapp.c](#).  
*Posted 04/18/2013. Sharbel Bousemaan*
- p. 872, Figure 10.7, `rio_readnb`, line 9: The check for `EINTR` is redundant because it is already handled by the call to `rio_read` in the previous line. [Updated version of csapp.c](#).  
*Posted 09/19/2012. Seth Chaiken*

- **Chapter 11: Network Programming**

- p. 888, paragraph 5, last line. "bridge C's segment" should be "host C's segment".  
*Posted 11/27/2010. Steven Tricanowicz*
- p. 917, line 2. The lecture notes present a version of `adder` that uses a slight different interface for its arguments: `n1=val1&n2=val2`, as opposed to `val1&val2`.  
*Posted 04/18/2013. Michael Spear*

- **Chapter 12: Concurrent Programming**

- p. 940, last line, "`read_set`" should be "`ready_set`".  
*Posted 04/6/2010. David O'Hallaron*
- p. 950, Sec 12.3.4, `pthread_exit` prototype. "Returns: 0 if OK, nonzero on error" should be "Never returns".  
*Posted 11/27/2010. Steven Tricanowicz*
- p. 951, Sec 12.3.5, last paragraph, line 2, "`pthread_wait`" should be "`pthread_join`".  
*Posted 4/19/2011. Yang Firo*
- p. 963, definition of `P(s)`, line 2. "process is restarted" should be "thread is restarted"  
*Posted 07/23/2010. Yili Gong*
- p. 983, para 3. The race is actually between the next increment of `i` in line 11 (not the next call to `pthread_create` in line 12), and the dereferencing and assignment of `vargp` in line 21.  
*Posted 05/16/2011. Shuang Yang*
- p. 984, Practice Problem 12.13. "immediately after line 15" should be "immediately after line 14".  
*Posted 07/30/2012. Yili Gong*
- p. 987. The simple *mutex lock ordering rule* presented here is not strong enough to prevent all circular deadlocks. E.g.,

```
init: r=s=t=1
thread 1: P(r)P(s)V(s)V(r)
thread 2: P(s)P(t)V(t)V(s)
thread 3: P(t)P(r)V(r)V(t)
```

Replace with the following stronger rule, due to Dijkstra: *Assign a total ordering to all mutexes. Each thread acquires its mutexes in order, and releases them in reverse order.*

*Posted 05/21/2012. Cheng Chen*

- p. 996, Solution to Problem 12.7, step 10, the values of `%eax1` and `%eax2` should be "-" and "1" respectively.  
*Posted 4/19/2011. Yang Firo*
- p. 997, Solution to Practice Problem 12.13. "line 15" should be "line 14" and "line 25" should be "line 24".  
*Posted 07/30/2012. Yili Gong*
- p. 997, Solution to Problem 12.12, line 2. "`gethostbyname`" should be "`ctime`".  
*Posted 4/19/2011. Yang Firo*

- **Appendix A: Error Handling**

- **Index**