

页码	页码行号	原文	修改后	备注
009	P009 1.5节第四行	world/n	world\n	第8次印刷已修订
011	P011 旁注第4行	Mutics	Mu <b>l</b> tics	第8次印刷已修订
026	P026 练习题2.2	给出2的不同次幂的二进制和十六进制表示	给出2的不同次幂的 <b>十进制</b> 和十六进制表示	第8次印刷已修订
029	P029 正文倒数第二段最后一行	IOS	<b>iOS</b>	第3次印刷已修订
044	P44 第7行	图2-11展示的是	图2- <b>12</b> 展示的是	
046	P046 图2-14下第一段第5行	也就意味着能表示的整数比负数少一个	也就意味着能表示的 <b>正数</b> 比负数少一个	第8次印刷已修订
050	P050 第2行	除了参数是无符号的，而结果是以补码表示的	<b>除了参数是以补码表示的，而结果是无符号的</b>	
050	P050 第3行	生成一个数的无符号表示和x的补码表示相同	<b>生成一个数，这个数的补码表示和x的无符号表示相同</b>	
052	P052 第一段最后一行	65 563	<b>65 536</b>	第8次印刷已修订
054	P054 第一个原理中	宽度为w'的位向量	宽度为 <b>w</b> 的位向量	第3次印刷已修订
054	P054 原理：补码数的符号扩展	第二个“宽度为w”	宽度为 <b>w'</b>	第8次印刷已修订
061	P061 图2-22	$x + ^u_y$	$x + ^b_y$	第8次印刷已修订
065	P65 第9行	或者 $x + y < TMax_w$	<b>否则</b> $x + y < TMax_w$	
065	P65 练习题2.32第2行	假设你写的练习题2.30的代码如下所示：	完成练习题2.30的代码后，你写下如下代码：	
067	P106 练习题2.46答案A中		小数点和1100之间有25个0	第8次印刷已修订
067	P067 2.3.5第3行		第一个负号删除	第8次印刷已修订
068	P068 第7行	$x' *_w y'$	$x' *_w y'$	
073	P073 中间，原理：除以2的幂的补码除法，向上舍入	$\lfloor x/2^k \rfloor$	<b><math>\lceil x/2^k \rceil</math></b>	第3次印刷已修订
074	P074 第2行	$\lfloor x/2^k \rfloor$	<b><math>\lceil x/2^k \rceil</math></b>	第8次印刷已修订
079	P79 旁注下第2段	逐渐溢出	逐渐 <b>下溢</b>	
082	P082	阶码字段除了最高有效位等于1以外，其他位都等于0	阶码字段除了最高有效位等于 <b>0</b> 以外，所有其他位都等于 <b>1</b>	第8次印刷已修订
086	P086 最后一行	-21483648	<b>-2147483648</b>	第8次印刷已修订

页码	页码行号	原文	修改后	备注
102	P102 练习题2.24 答案最后一行	-4 ~ 1序列的反复	-4 ~ <b>-1</b> 序列的反复	
103	P103 练习题2.32 答案	实际上, 情况恰恰相反: 当x为负数时, tsub_ok(x, TMin)为1, 而 当x为非负时, 它为0。	实际上, 情况应该恰恰 相反才对: 当x为负数 时, tsub_ok(x, TMin)应 该为1, 而当x为非负 时, 它应该为0。	第8次印刷已修订
104	P104 练习题2.35 答案3) 第二行	$2^w$	$2^{w-1}$ 说明: 原题只需要 $ r  < 2^w$ , 所以原来的描述 和证明仍然是正确的。	
104	P104 2.35 1) 最后 一行		第二个“乘法不会溢出”改 为“乘法会溢出”	
106	P073 图2-29标题 中	除了是向下舍入	除 <b>法</b> 是向下舍入	第8次印刷已修订
106	P106 2.44 F.	补码和无符号乘法有相同的 位级行为	补码和无符号 <b>加法</b> 有相 同的位级行为	
107	P107 练习题2.51 答案B.中		<b>x' - 0.1的二进制表示中小 数点后应有25个0</b>	第8次印刷已修订
108	P108 练习题2.54 答案B部分中	(int)(double)x	(int)( <b>float</b> )x	第8次印刷已修订
114	P114 旁注上倒数 第二段倒数第三行	64TB	256TB	第8次印刷已修订
119	P119 3.4节第一段	%bp, %ebp, %rbp	<b>%sp, %esp, %rsp</b>	第8次印刷已修订
120	P120 倒数第2行	有些程序会明确地读写	有些 <b>指令</b> 会明确地读写	
124	P124 旁注第2行末 尾		将“ - ”和下一行开头的“1 ”合并到一起	第3次印刷已修订
129	P129 图3-10标题 说明文字最后一句		<b>注意ATT格式的汇编代 码中操作数的顺序与一 般的直觉相反</b>	第8次印刷已修订
130	P130 练习题3.6, 第一个表达式中	%ax	<b>%rax</b>	第3次印刷已修订
133	P133 图3-12中第 三行	clto	<b>cqto</b>	第3次印刷已修订
133	P133 图3-12中 idivq和divq两条指 令的第二行中	R[%rdx]<- R[%rdx]:R[%rax]÷S	R[ <b>%rax</b> ]<- R[%rdx]:R[%rax]÷S	第3次印刷已修订
133	P133 图3-12最后 一行	R[%rdx]← R[%rdx]:R[%rax]÷S	<b>R[%rax]</b> ← R[%rdx]:R[%rax]÷S	

页码	页码行号	原文	修改后	备注
134	P134 倒数第4段	对大多数64位除法应用来说，除数也常常....	对大多数64位除法应用来说， <b>被除数</b> 也常常....	第3次印刷已修订
138	P138 练习题3.13 D中	setne %a	setne %a <b>l</b>	第8次印刷已修订
145	P145 图3-17(c)代码第8行	return tval	return <b>r</b> val	第8次印刷已修订
147	P147 倒数第3行	vt的值	<b>v</b> e的值	第8次印刷已修订
164	P164 最后一行	stack fram	stack <b>frame</b>	第8次印刷已修订
166	P166 最后一行	T1 ~ T4 (main中)	T1 ~ T4 ( <b>top</b> ) 中	第8次印刷已修订
167	P167 图3-27		T1的PC应为0x4005 <b>45</b> ，T2的PC应为0x4005 <b>49</b>	第3次印刷已修订
168	P168 3.7.3节第二段第一行	(例如整数和指针)	( <b>即</b> 整数和指针)	第8次印刷已修订
168	P138 练习题3.14 D.	setne	set <b>l</b> e	第8次印刷已修订
169	P169 图3-29上最后一行	每个都是8字节的	每个 <b>指针</b> 都是8字节的	第3次印刷已修订
173	P173 第2段第1行		“ <b>被</b> 调用者保存”中的“ <b>被</b> ”字也应用特殊字体标注，避免误读	第3次印刷已修订
173	P173 第8行	保存的寄存器	<b>被</b> 保存的寄存器	第8次印刷已修订
176	P176 表格下一行	每个需要8个字节	每个需要 <b>4</b> 个字节	第8次印刷已修订
177	P177 3.8.2节中的表格第三行第四列	movl (%rdx), %rax	movl (%rdx), <b>%eax</b>	第8次印刷已修订
177	P177 3.8.2节中的表格第6行第4列	leaq-4(%rdx,%rcx,4),%rax	<b>leaq -4</b> (%rdx,%rcx,4),%rax	
177	P177 3.8.2节中的表格第7行第4列	movl-12(%rdx,%rcx,4),%eax	<b>movl -4</b> (%rdx,%rcx,4),%eax	
196	P196 练习题3.46 A部分最后一行	0x3 ~ 0x39	0x <b>30</b> ~ 0x39	第8次印刷已修订
202	P202 第5行	在执行工程中	在执行 <b>过程</b> 中	第8次印刷已修订
208	P208 练习题3.50上倒数第4行	因为它们既不是整数也不是指针	因为它们 <b>是整数或者指针</b>	第8次印刷已修订
212	P212 图3-50第2行第2列	vorpd	<b>x</b> orpd	

页码	页码行号	原文	修改后	备注
213	P213 3.11.6下表中	ucomiss和ucomisd	<b>vucomiss和vucomisd</b> 说明: vucomiss和vucomisd是AVX指令, ucomiss和ucomisd是SSE指令	
216	P216 第1行	参见原书546页	参见 <b>本书376</b> 页	第8次印刷已修订
226	P226 最后一行		这些例子不依赖整型提升的规则。这些规则与对比int更小的数据值实施的操作有关, 可以参考 <a href="https://wiki.sei.cmu.edu/confluence/display/c/INT02-C.+Understand+integer+conversion+rules">https://wiki.sei.cmu.edu/confluence/display/c/INT02-C.+Understand+integer+conversion+rules</a> 。本书中没有包含这部分内容。	
227	P227 练习题3.4的答案, 表格第三行	novsbl	<b>movsbl</b>	第3次印刷已修订
232	P232 练习题3.22的答案中	直到20!才溢出	<b>到20!都不会溢出</b>	第8次印刷已修订
234	P234 练习题3.30答案中	在switch语句体中缺失了情况标号3和-6	在switch语句体中缺失了情况标号3和 <b>6</b>	第8次印刷已修订
235	P235 练习题3.33, 第3行	v+=b	<b>*v+=b</b>	第3次印刷已修订
236	P236 3.37答案的表格中		所有的leal都应 <b>为leaq</b>	第8次印刷已修订
236	P236 练习题3.37答案		表格第四列中所有leal都应该 <b>为leaq</b>	
238	P238 3.43答案的第5行	addq \$, %rdi	addq <b>\$10</b> , %rdi	第8次印刷已修订
238	P238 3.43答案的第7行	movq %rdi, %rsi	movq %rdi, <b>(%rsi)</b>	第8次印刷已修订
240	P240 练习题3.49答案A中	这个值减去s1就得到s2	<b>s1减去这个值</b> 就得到s2	第3次印刷已修订
240	P240 3.49 B.	舍入到最近的8的倍数	<b>向上</b> 舍入到最近的8的倍数	第8次印刷已修订
240	P240 练习题3.51答案, 表格第6行第3列	vctsi2ssq	vc <b>v</b> tsi2ssq	
247	P247 倒数第3行	附加的4字节	附加的 <b>8</b> 字节	第8次印刷已修订
251	P251 图4-6第9行注释	Test sum	Test <b>count</b>	第8次印刷已修订

页码	页码行号	原文	修改后	备注
251	P251 图4-6 x86-64 代码第9行注释	Test sum	Test <b>count</b>	
253	P253 第2段	0x000d000d000d000d, 0x00c000c000c000c0, 0x0b000b00b0bb0b00, 0xa000a000a000a000	0x000d000d000d, 0x00c000c000c0, 0x0b000b000b00, 0xa000a000a000	第8次印刷已修订
259	P259 练习题4.10 第2行	需要64个字级的异或电路	需要64个 <b>位</b> 级的异或电路	第8次印刷已修订
264	P264 倒数第3段第4行	取出一个四字节常数字	取出一个 <b>8</b> 字节常数字	
264	P264 第一段第6行	对于读操作来说	<b>与读操作一样</b>	
267	P267 图4-19 访存一行第3列	$valE \leftarrow M_8[valE]$	<b>valM</b> $\leftarrow M_8[valE]$	
268	P268 图4-20第3列 倒数第4行	valE	val <b>M</b>	第8次印刷已修订
268	P268 旁注中的图, “取指”行“通用”栏第3行	$valP \leftarrow M_8[PC+2]$	val <b>C</b> $\leftarrow M_8[PC+2]$	第8次印刷已修订
270	P270 练习题4.17 第一行	rmmovq	<b>rrmovq</b>	
275	P275 倒数第二行	组合逻辑被条件码寄存器环绕着	组合逻辑 <b>环绕着</b> 条件码寄存器	第8次印刷已修订
276	P276 图4-25 (1) 和 (2) 中	CC 000	CC <b>100</b>	第8次印刷已修订
276	P276 图4-25的表格第2行	irmovq \$0x200, %rbx	irmovq \$0x200, <b>%rdx</b>	第8次印刷已修订
276	P276 图4-25表格第1行	\$sx100	<b>\$0</b> x100	第8次印刷已修订
276	P276 图4-25	程序计算器	<b>程序计数器</b>	
277	P277 第7个指令	IOPL	IOP <b>Q</b>	第8次印刷已修订
288	P288 4.5节上最后一行	是不可接收的	是不可接 <b>受</b> 的	第8次印刷已修订
288	P288 图4-38上倒数第二行	如图4-38c所示	如图4-38 <b>d</b> 所示	
293	P293 4.5.4节上倒数第2段倒数第2行	标号为“Data”的块	标号为“ <b>数据</b> ”的块	第8次印刷已修订
296	P296 图4-44的说明中	直到周期7结束时	直到周期7 <b>开始</b> 时	第8次印刷已修订

页码	页码行号	原文	修改后	备注
306	P306 倒数第一段		说明：我们的处理器设计在异常发生时，不会试图将程序计数器（PC）设置为准确的值。在更完备的实现中，PC会被设置为导致异常的指令的地址，但是流水线化的实现并不是这样。	
316	P316 图4-63 左上角代码第二行空白处		<b>0x00a: xorq %rsp,%rsp #CC = 100</b>	第8次印刷已修订
317	P317 图4-65 b) 暂停	暂停=0	暂停= <b>1</b>	第8次印刷已修订
317	P317 图4-65 第三幅图	b) 气泡	<b>c)</b> 气泡	第8次印刷已修订
319	P319 练习题4.38 图第4行第1列	预测错误的分支	<b>加载/使用</b>	第8次印刷已修订
335	P335 练习题4.14 答案表格中，写回一行具体一列中	R[%rsp] <-- 9	R[ <b>%rax</b> ] <-- 9	第8次印刷已修订
367	P367 倒数第二段 倒数第二行	图5-17给出了当循环展开到10次时	图5-17给出了当循环展开到 <b>大约</b> 10次时	第8次印刷已修订
368	P368 第一段第四行	将data[i]加到acc上，第二条将data[i+1]加到acc上	将data[i] <b>乘到</b> acc上，第二条将data[i+1] <b>乘到</b> acc上	第8次印刷已修订
374	P374 图5-27右侧代码第2、3行	vmovsd	<b>vmulsd</b>	第8次印刷已修订
377	P377 网络旁注中配图		第2行最后两列的long和int应该分别为 <b>float</b> 和 <b>double</b>	第8次印刷已修订
386	P386 倒数第3行	只有到计算出加载和存储的地址被计算出来以后	只有到加载和存储的地址被计算出来以后	
393	P393 参考文献说明，第二段第一行	形式化描述了编辑器可以产生更有效代码的方法	形式化描述了 <b>编译器</b> 可以产生更有效代码的方法	第8次印刷已修订
395	P395 倒数第3行	变量x	变量 <b>acc</b>	第8次印刷已修订
404	P404 第8行	DDR (8位)	DDR <b>3</b> (8位)	第8次印刷已修订
406	P406 6.1.2节上倒数第3行	内存从内存总线读出地址	<b>主存</b> 从内存总线读出地址	

页码	页码行号	原文	修改后	备注
409	P409 图6-10下第一段第3行	绕地球一天	绕地球 <b>一周</b>	第8次印刷已修订
412	P412 旁注第4行	VCEe	PC <b>I</b> e	第4次印刷已更正
412	P412 第一段	与SATA适配器不同，它只能支持一个驱动器	<b>而SATA适配器与之不同，只能支持一个驱动器</b>	第8次印刷已修订
420	P420 6.2.3第4行最后	步长为l	步长为 <b>1</b>	第3次印刷已修订
439	P439 第9行		去掉“50个周期；”后的“，”	第8次印刷已修订
439	P439 倒数第5行	较低层上使用比较小的相联度	较低层上使用比较 <b>大</b> 的相联度	第8次印刷已修订
445	P445 图6-41上面一段第二行	128KB	128 <b>MB</b>	第8次印刷已修订
448	P448 图6-44 f)	$C[i][j] += A[i][k] * r$	$C[i][j] += \mathbf{B[k][j]} * r$	第8次印刷已修订
479	P479 图7-9下第三行	回想一下3.6.3节	回想一下 <b>3.6.4</b> 节	
480	P480 图7-11上一行	objdump-dx main.o	<b>objdump -dx</b> main.o	
498	P498 作业7.8 A.中	static int main=1[	static int main=1 <b>;</b>	第8次印刷已修订
498	P498 作业7.8第一行	REF(x, i)→DEF(x, k)	REF( <b>x.i</b> )→DEF( <b>x.k</b> )	
499	P499 练习题7.1答案中	buf在mina.o中定义	buf在 <b>m.o</b> 中定义	第8次印刷已修订
503	P503 图8-3	x 84	<b>x 8</b>	第8次印刷已修订
515	P515 图8-16中 x==1		<b>x=1</b>	
523	P523 getenv函数描述中返回值解释	若存在则为指向name的指针	若存在则为指向 <b>value</b> 的指针	
532	P532 图8-30第15行	pause();	<b>P</b> ause();	
540	P540 8.5.6上一行	实现	<b>使用</b>	
557	P557 图8-49		右边的1改为 <b>2</b>	第3次印刷已修订
566	P566 第8行	V P1	<b>VP 1</b>	第8次印刷已修订
566	P566 图9-9		进程i与进程j的虚拟地址空间之间有"..."	
570	P570 第3行	图9-13b	图9-13 <b>a</b>	



页码	页码行号	原文	修改后	备注
572	P572 倒数第一行	对于只有一级的页表结构，PPO和VPO是相同的	<b>和只有一级到页表结构一样</b> ，PPO和VPO是相同的	
574	P574 图9-20	第二幅图最左边表头第一个字“位”	<b>组</b>	
574	P574 最上面虚拟地址的划分		TLBI的左右箭头分别位于7和8之间以及5和6之间；TLBT的右箭头位于7和8之间	
579	P579 第3段第3行	提供到一个L1 PET的偏移量	提供到一个L1 <b>PTE</b> 的偏移量	
580	P580 图9-26	0x40000000	<b>0x400000</b>	第8次印刷已修订
580	P580 图9-26		地址空间的内核部分对每个进程来说都一样。不存在对每个进程来说不相同的内核虚拟内存	
591	P591 倒数第二行	请求要求6个字	请求要求 <b>8</b> 个字	
599	P599 图9-43下第二行	viod*	<b>void *</b>	第8次印刷已修订
608	P608 图9-52下第一行	其中每个块都是未分配的	其中每个块都是 <b>未标记的</b>	第8次印刷已修订
626	P626 旁注下第3行	只含有20多个字节	只含有20个字节	
660	P660 练习题11.4第三行	inet_pton	<b>inet_ntop</b>	第8次印刷已修订
668	P668 "2.HTTP响应"上第四行	使得代理链中的代理能够判断它是否可以在本地缓存中拥有一个被请求内容的副本	使得代理链中的代理能够判断它是否 <b>在本地缓存中已经拥有被请求内容的副本了</b>	第8次印刷已修订
671	P671 倒数第4行	图11-8	图 <b>10</b> -8	第8次印刷已修订
698	P698 练习题12.6表格倒数第二行	myid.po	myid.p <b>0</b>	第3次印刷已修订
700	P700 图12-17, Ui对应的行	addq %eax	addq <b>\$1, %rdx</b>	第8次印刷已修订
700	P700 图12-17, Si对应的行	movq %eax, cnt(%rip)	movq <b>%rdx</b> , cnt(%rip)	第8次印刷已修订
713	P713 第二段第一句话	程序单线程顺序运行非常慢，几乎比多线程并行运行时慢了一个数量级	<b>这个程序不仅单线程顺序运行时非常慢，多线程并行运行比单线程几乎还要慢一个数量级</b>	



页码	页码行号	原文	修改后	备注
727	P727 练习题12.12 答案	gethostbyname	<b>ctime</b>	第3次印刷已修订
727	P727 12.12中	gethostbyname函数	<b>ctime</b> 函数	第4次印刷已更正