

2023-2학기 기본교재 『C++프로그래밍』 정오표

쪽	행	틀린 내용	바로 잡은 내용
53쪽	소스코드2-10 7행	struct C2dType // 2차원 좌표 구조체	struct C2dType { // 2차원 좌표 구조체
71쪽	소스코드2-13 11행	while (*pt) // 문자열의 ...	while (*pt) { // 문자열의 ...
95쪽	소스코드3-8 16행	{	}
138쪽	16행	: maxValuec.maxValue, valuec.value	: maxValue {c.maxValue}, value {c.value} { }
139쪽	표4-6	VecF(int d, float* a)	VecF(int d, const float* a)
140쪽	소스코드4-11 11행	VecF(int d, float* ...	VecF(int d, const float* ...
141쪽	소스코드4-12 결과	[1 2 4] + [1 2 3]	[1 2 3] [1 2 3]
145쪽	밑에서 2행	r-value 중 곧 소멸될 예정인 객체를 x-value(expiring value)라고 ...	r-value 중 l-value처럼 실체가 존재하지만 곧 소멸될 예정 등의 이유로 가지고 있는 자 원을 다른 객체로 이동할 수 있는 것을 x-value(eXpiring l-value)라고 ...
146쪽	9행	... // 상수 l-value로는 ...	... // 상수 l-value 참조로는 ...
146쪽	11행	... r-value(x-value)임	... r-value임
147쪽	소스코드4-15 15행	VecF(const VecF& fv) : n fv.n { }	VecF(const VecF& fv) : n { fv.n } { }
156쪽	표4-10 마지막 행	char chkFull() 스택이 가득 차 있는지 검사함	char pop() 스택에서 데이터를 꺼냄
163쪽	소스코드4-22 7행	... rPart(r), iPart(i)	... rPart(r), iPart(i) { }
170쪽	3행	② x-value는 r-value 중 ...	② l-value는 r-value 중 ...
	4행	... 참조 대상 임시 객체가 ...	... 참조 대상 객체가 ...
177쪽	소스코드5-1 9행	... const return a;	... const { return a; }
183쪽	소스코드5-6 11행	return Comlex1(rPart, -iPart);	return Comlex2(rPart, -iPart);
193쪽	소스코드5-14 8행	cout << ")";	os << ")";
198쪽	소스코드5-18 20행	cout << "v1 = " << v1 << ", ";	cout << "--> v1 = " << v1 << ", ";
200쪽	소스코드5-20 12행	... // i번째 원소의 ...	... // i번 원소의 ...
	17행	... // i번째 원소의 값 반환	... // i번 원소 반환

쪽	행	틀린 내용	바로 잡은 내용																				
202쪽	소스코드5-21 15행	... // i번 째 원소를 ...	... // i번 원소를 ...																				
	21행	... // i번 째 원소 의 값 반환	... // i번 원소 반환																				
	28행	... // i번 째 원소 의 값 반환	... // i번 원소 반환																				
208쪽	소스코드5-25 5행	... : len(0), bufSize(s) {	... : len(s), bufSize(s) {																				
	7행	buf[0] = '\0';	buf[s] = '\0';																				
211쪽	소스코드5-25 83행	bufSize = len += mstr.len;	행 삭제, 84~89행의 번호를 83~88로 변경																				
	변경된 83행	... new char[len + 1];	... new char[(bufSize = len + mstr.len) + 1];																				
	변경된 88행	str cat (buf, mstr.buf);	str cpy (buf + len , mstr.buf);																				
	새로운 89행		len += mstr.len;																				
213쪽	밑에서 4행	... 37 행에서 ...	... 47 행에서 ...																				
233쪽	밑에서 2행	... [소스코드 6-2] 15 행이나 ...	... [소스코드 6-2] 14 행이나 ...																				
263쪽	밑에서 12행	28 행은 프로그램에서 ...	32 행은 프로그램에서 ...																				
267쪽	소스코드6-17 15행	:cx(x), cy(y), radius(r)	:cx(x), cy(y), radius(r) { }																				
277쪽	밑에서 3행	... [예제 6- 4]에서 2개의 ...	... [예제 6- 5]에서 2개의 ...																				
286쪽	소스코드6-34 10행	Parttime(s const string& n, ...	Parttime(const string& n, ...																				
307쪽	그림 7-1	(라) i= 1 일 때의 반복 후 ... (마) i= 1 일 때의 반복 후 ... i=3<table><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr></table> ※ (마)의 그림에서 8과 9의 배경색 제거	0	1	2	3	4	5	6	7	8	9	(라) i= 2 일 때의 반복 후 ... (마) i= 3 일 때의 반복 후 ... i=3<table><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr></table>	0	1	2	3	4	5	6	7	8	9
		0	1	2	3	4	5	6	7	8	9												
0	1	2	3	4	5	6	7	8	9														
308쪽	3행	... 11행에서 'j < size - 1 ').	... 11행에서 'j < size - i ').																				
308쪽	소스코드7-8 11행	cout << x[i] << " ";	cout << i << " ";																				
322쪽	소스코드7-11 25행	cout << "벡터2 에 저장된 값 : ";	cout << "벡터2 : ";																				
329쪽	소스코드7-13 18행	{ "박영철", "010-2233-4455" },	{ "박영철", "010-2233-4455" }																				
	소스코드7-13 23행	pBook.insert("박영철","010-2233-4455");	pBook.insert({ "박영철","010-2233-4455" });																				
349쪽	소스코드8-8 17행	class BadIndex; // exception class	class BadIndex { ; // exception class																				
353쪽	밑에서 7행	const char* what() const return ...	const char* what() const { return ...																				
355쪽	2행	f()를 호출하는데, 만일 g ()에서 ...	f()를 호출하는데, 만일 f ()에서 ...																				
359쪽	밑에서 2행	... [소스코드 8- 11]의 [] 연산자 ...	... [소스코드 8- 4]의 [] 연산자 ...																				
384쪽	밑에서 4행	... 6 행에서 이전의 실수 표시 ...	... 5 행에서 이전의 실수 표시 ...																				