

Submission Date:	Late Penalty	
	Total Deductions from below	
	Total Score	
Design and Engineering of the Array Template (-30 points maximum) (Have the student show you the relevant areas in his/her implementation.)		Category Deduction
Template data members: must have array pointer, dimension and a way to track usage -5 each all data members must be private -10 Operation: must do ordered insertions -10 Template dynamic memory management: constructor must allocate physical array dynamically -15 destructor must deallocate physical array -10 Deep copy issues: must have a deep copy constructor -5 must have an assignment overload -5		
Design and Engineering of the Book Class (-10 points maximum)		
Data members: ISBN, title, category, author ID, sales, year -3 each all data members must be private -5 each Overloaded operators: must overload at least two relational operators, e.g. == and < -5 each		
Design and Engineering of the Database Class (-15 points maximum)		
Data members: must have an association or aggregation with two array objects -5 each Must have public function members that support: adding books and authors -5 per line listing books and authors updating sales for a book finding an author saving book and author lists to disk		
Comments:		
Internal Documentation (-10 points maximum)		
Spot-check comments in a randomly selected pair of class source/header files: Class description (header block) -5 Function/method headers: -5 (statement of purpose, parameter comments, pre/post conditions)		
Comments:		

Correctness of Program Operation	(-35 points maximum)	
General quality of log file:	(-5 points maximum)	
Commands should be echoed to log file	-5	
Confirmation/error messages should be logged for each command	-5	
Command output should be delimited for easy reading	-3	
Loading initial author and book dB info:	(-15 points maximum)	
Correct number of books (20)	-5	
Correct book at index 15 (The Saxon Shore -- 0-812-54416-1)	-5	
Correct number of authors (12)	-5	
Correct author at index 9 (Tolkien, J R R -- T0001)	-5	
Simple dB updates and searches:	(-10 points maximum)	
Should delete book 0-06-102069-9 (Equal Rites)	-5	
Should add book 0-0022-2155-1 (The Dark Tower, . . .)	-5	
(Verify by checking the author listings. Deduct full credit for each if author search doesn't work.)		
Problem dB updates:	(-10 points maximum)	
Should not add book (1-019-10560-X) without matching author entry	-5	
Should not delete book (0-808-56648-2) that's not in dB	-5	
Should not add duplicate author (Burke)	-5	
(Must produce a relevant error message to receive credit.)		
Searching:	(-8 points maximum)	
Author search on B0001 (Burke)	-4 for each missing or incorrect entry	
Overflow detection:	(-10 points maximum)	
Should refuse to insert book 0-380-72121-X (In the Electric Mist . . .)	-5 points	
Should insert author Gaimon, Neil	-5 points	
Should refuse to insert author Gardner, John	-5 points	
(Must produce a relevant error message to receive credit.)		
Final book list and author search:	(- 10 points maximum)	
Correct number of books (19)	-5	
Correct book at index 8 (Look Homeward Angel)	-5	
Correct info for final author search	-5	
Comments:		
Other Adjustments		
Build problems: in the event of compile or link errors, the total score is not to exceed 30%.		
Runtime problems		
Runtime exception, program hangs during execution, error on exit	-5	

Bookstore dB Grading Procedure

Checksheets and Demos:

Since you'll be doing demos for this project, it's probably simplest if you just fill out a hardcopy checksheet for each demo, and then later fill in an electronic copy for our records. You should fill out one checksheet for each submitted project that is demonstrated.

Make comments! There are comment fields on the checksheet template to record anything unusual or important that is not clearly indicated otherwise. If you don't write it down, we won't know about it.

If a student made more than one submission, score the last one unless the student requests you score an earlier submission. If a file doesn't open properly, notify us of the problem, assign a score of zero and we'll decide what to do. If a student omitted a necessary source file:

- for a missing header file (NOT including a template implementation), have the student supply the missing file and deduct 10 points.
- for a missing source file (including a template implementation even if it's a .h file), have the student supply the missing file and apply the late penalty as if the project had been submitted at the time of the demo.

Record the scores in the spreadsheet provided by the instructor. If there are submissions from students who are not in the spreadsheet, record their names and scores at the bottom of the list. Record zeros for students who did not make a submission or did not make a demo.

When you're done, return the completed checksheets, and the backup copies, to us.

The checksheet is pretty simple. Check each listed item and show any deductions you make for each item in the right column. If there's anything special to note, do so in the comments section. This is particularly important if you have any major issues of disagreement with the student, or if there's something unusual. **If you don't make a note about it, we won't have any record of it.**

For the design and engineering sections, ask the student to show you the things you need to see.

For documentation, examine the students' code as he/she shows you the design/implementation details listed in the first section of the checksheet; you **should** assign partial credit here unless they have NO internal documentation.

For the program correctness section, have the student build an executable and then run it on the supplied input test file. If the program doesn't build, they lose 70 points here. If it builds but doesn't produce any useful output, they lose 35 points here. Otherwise, have them open their output file and compare it to the given correct output. There is not much ground for partial credit on each individual item. Note that there are many output items shown in the log file that don't have point values assigned to them.

Under Other Adjustments, assign penalties only if there are runtime errors but substantial gradable output was produced.

Late penalty:

Be sure to take the submission timestamp from the submit log file, not from the FTP client or file timestamp.

Monday	April 2	1 day late	20% off	
Tuesday	April 3	2 days late	40% off	
After	April 3	>2 days late	50% off	(make this the <u>maximum</u> late penalty)

11365	McQuain	10:10 MWF
11366	Barnette	12:30 TTh
11367	McQuain	8:00 MWF
11368	Helms	3:30 TTh

If something comes up you can't figure out, let us know.