

CS 159 – HW #02

5 Points Possible

Due: February 6, 2012 at 11:00pm.

Problem: Another college-level class is graded based ONLY on exam performance. There are a total of four exams in this course with the first three being worth 100 points each and a fourth exam worth 150 points. The scale for this class is 85% required to earn an A, 75% for B, 65% for C, 55% for D.

Given four integers as input display the score for each exam, the grade earned on each exam, and the total and final grade for the course.

Example Execution #1:

```
Enter score for exam #1: 55
Enter score for exam #2: 70
Enter score for exam #3: 65
Enter score for exam #4: 100
```

Exam	Score	Grade
1	55	D
2	70	C
3	65	C
4	100	C

Total: 290
Grade: D

Example Execution #3:

```
Enter score for exam #1: 60
Enter score for exam #2: 65
Enter score for exam #3: 70
Enter score for exam #4: 98
```

Exam	Score	Grade
1	60	D
2	65	C
3	70	C
4	98	C

Total: 293
Grade: C

Example Execution #2:

```
Enter score for exam #1: 84
Enter score for exam #2: 85
Enter score for exam #3: 86
Enter score for exam #4: 126
```

Exam	Score	Grade
1	84	B
2	85	A
3	86	A
4	126	B

Total: 381
Grade: B

Example Execution #4:

```
Enter score for exam #1: 94
Enter score for exam #2: 96
Enter score for exam #3: 85
Enter score for exam #4: 127
```

Exam	Score	Grade
1	94	A
2	96	A
3	85	A
4	127	B

Total: 402
Grade: A

Additional Very Important Requirements and Reminders:

- Accept the input of four integers and produce output formatted **exactly** as seen in the example executions above.
 - You may assume the user will always enter meaningful integer data when testing your program.
 - Do not add any “bonus” features not demonstrated in the example executions provided.
- **DO NOT** use any material found outside of the first FOUR chapters of the C text.
 - **The use of any SELECTION constructs from Chapter 5 or any other logical operator will result in a zero score for this assignment.**
- Additionally, **this assignment MUST make good use of user-defined functions to be considered for partial credit.**
- A program **MUST** compile to be considered for partial credit. The submission script will reject the submission of any file that does not compile and is therefore untestable.

Academic Integrity Reminder:

- Please review the official policies and consequences of the course as they relate to academic integrity. The assignment you submit should be your own original work. You should be consulting only course staff regarding your specific algorithm for assistance. Collaboration is not permitted on individual homework assignments.

Course Programming and Documentation Standards Reminders:

- Make use of the course FUNCTION HEADER FILE (`head_fx`) with every user-defined function in your program.
 - All parameters to a function must be listed and commented, one per line, in the course function header. Other locally declared variables will be commented to the right of each declaration statement.
 - Details regarding the course expectation of user-defined function use can be found both in the standards document and chapter 4 notes.
- Place a single space between all operators and operands.
- Comment all variables to the right of each declaration. Declare only one variable per line.
 - **Note that several programs in the text use a single line comment to indicate the start of the local declaration and executable statement sections of the `main` function.** At no point during the semester should these two sections of a function be permitted to overlap. You may consider adopting this habit of commenting the start of each section (see program 2-7 on page 71-72 of the C programming text as an example) to help you avoid this mistake.
- Select meaningful identifiers (names) for all variables and symbolic/defined constants in your program.
- Indent all code found within the `main` function **exactly** two spaces.
- Do not single (or double) space the entire program, use blank lines when appropriate.

When you submit... only the last attempt of a submission is kept for grading. All other submissions are overwritten and cannot be recovered. You may make multiple submissions but only the last attempt is retained and graded.

- Verify in the e-mail sent to you by the course that you have submitted the correct file, to the correct assignment (`hw02`), and to the correct lab section. Forwarding confirmation e-mails from Purdue to external e-mail services may result in the mail being undelivered or end up being identified as spam.
- Leave time prior to the due date to seek assistance should you experience difficulties completing or submitting this assignment.
- All attempts to submit via a method other than through the sage server as set up during the first week of the semester will be denied consideration. Leave time to seek assistance should you struggle to use any of the tools of the course.

Assignment deadlines... are firm and the electronic submission will disable promptly as advertised. We can only grade what you submit as expected prior to the assignment deadline.

All course programming and documentation standards are in effect for this and each assignment this semester. Please review this document in your course notes packet.
