CS 142 Midterm 1

Version 0.6

Instructor: R. P. Burton

Tuesday, February 12, 2013 to Friday, February 15, 2013, 6:00 p.m.

Late penalty is 20 points per day, weekend days included, advancing at 6:01 p.m. each day

Allowed materials include your CS 142 course text, your own notes, your own prior lab solutions, the CS 142 course website, and cplusplus.com

Disallowed materials include all other text resources, all other Internet

resources, and all neighbors (remember, everyone is thy neighbor).

**Instructions**

1. This midterm consists of a C++ programming problem. Read and understand the statement of the problem completely before beginning to design, code, and test. Understanding the problem correctly is part of the examination. If something seems unclear, ask a CS 142 TA (but no one else) for clarification.

2. As part of your design, consider test cases that will establish the correctness of your solution.

3. Produce a solution, which consists of *your* C++ code, with a comment at the beginning of your file which includes your name, your student ID number, and “CS 142 Winter 2013 Midterm 1.” Save your complete source file(s) using the online submission script. Attribute any code taken from or based on other sources (except for the course text and the course website). Code taken from or based on other sources is worth half credit.

4. You may pose questions to the CS 142 TAs at any time. However, the TAs generally are not permitted to answer questions related to design, C++ implementation, debugging, or testing.

5. Code well. This includes choosing good names for identifiers, avoiding magic numbers, choosing good control structures, formatting in a communicative way, etc. so that your code is reasonably self-documenting. Provide comments where required or appropriate, especially to help the TAs understand what your code is intended to do. Remember, you will be graded not only on whether your code (when entered, compiled, and executed) accomplishes the specified task, but also on how clearly and efficiently it accomplishes the specified task, and on how closely you adhere to the stated requirements and to good programming practices.

6. When you are finished, score your solution as indicated on the Scoring Sheet. fabs (your\_score – the\_TAs\_score) \* 0.5 – 5.0 will be subtracted from your score. Then go to the course website and follow the link labeled “Submit Exam” in the Exam Menu.

7. Sign here to request that your midterm be graded and to certify that no unfair information related to the midterm has been received by you, either directly or indirectly, and that none has been or will be conveyed by you. If we discover that you cheated or assisted someone in cheating, intentionally or unintentionally (including accidentally), your score for this exam may (and probably will) be rand() % 1. Discipline also may include Honor Code Office involvement, which can lead to loss of opportunities at Brigham Young University.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Name)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, 2013

(Signature) (Date)

**Hoop-dee-doo!**

**Background:** A *low tunnel* is a type of green house that is relatively *low* and *tunnel*-shaped. It might measure 4’ high x 3’ wide x 10’ long. Often the frame is made from 10’ lengths of metal conduit, some of which are bent into hoops and covered with protective fabric. Low tunnels extend the growing season by making it possible (a) to plant prior to the start of the normal growing season and (b) to keep plants alive after the end of the normal growing season; when plants are covered by low tunnels, they are protected from wind and freezing temperatures.



*A low tunnel frame made with metal conduit*

A typical residential “farmer” may have a plot of land that can be devoted to growing vegetables and flowers. Often this plot of land is rectangular. Ideally this plot of land could be covered entirely by low tunnels. However, low tunnels typically have rectangular footprints, measuring 3’ x 10’, 4’ x 10’, or 5’ x 10’, making complete coverage impossible. The hoops typically are formed by using a hoop bender to bend the conduit. Commercial hoop benders are available for 3’, 4’, and 5’ tunnel widths. To save the expense of purchasing multiple hoop benders, the residential farmer can be expected to opt for one width for all his low tunnels. Your help is needed to determine the optimum width (and hence the appropriate hoop bender), the number of tunnels, the percentage coverage, and the square footage covered, given the dimensions of the rectangular plot of land.

**Requirements:** Your task is to

* Prompt the residential farmer for the dimensions of his rectangular garden plot
* Store user input for length and width as integers
* Determine the low tunnel width and orientation that will maximize the coverage of his garden plot (and minimize the number of tunnels in the event of a tie in coverage; thirty 4’-wide low tunnels are better than forty 3’-wide low tunnels)
* Report the number of low tunnels, their common size and orientation , the percentage coverage of the garden plot by the low tunnels, and the square footage
* Allow the residential farmer to repeat the process for garden plots with different dimensions
* Provide clear and coherent output including all of the required prompts and results

**Sample Dialog** (user input is underlined):

What is the length of the garden plot? 83

What is the width of the garden plot? 24

You should make 48 low tunnels, 4’ x 10’ and place them parallel to the 83’ direction. Your coverage will be 96% (1920 square feet).

Another garden plot (y or n): y

What is the length of the garden plot? 9

What is the width of the garden plot? 9

Your garden plot is too small for low tunnels.

Another garden plot (y or n): y

.

.

.

Another garden plot (y or n): n

Good bye.

**Clarifications:**

* Assume a consistent low tunnel width and orientation for each garden plot (though they may be different for a garden plot of different dimensions).
* Reject any dimension provided by the user that is not a positive number.
* Report percentage coverage accurate to the nearest whole percentage.

**Hints:**

* Begin by making sure you understand the requirements. If you do not understand them, ask a CS 142 TA (but no one else) for clarification. A great solution to the wrong problem won’t earn many points. You can exceed the specified requirements, but you can’t modify or omit any of them.
* Before you write a single line of code, design your solution completely. Consider alternatives that may simplify the design and the coding. There are difficult ways to solve this problem that will take significant coding effort. Fortunately, there are also easy ways to solve this problem that will require a much smaller coding effort. Your design should be so complete that, given only a pencil, paper, a quiet place to work, and sufficient time, someone using your design could solve the problem without the aid of a computer. Determine test cases that can be used to verify the completeness and correctness of your solution.
* Implement your solution a piece at a time. The first version of your program might contain only one of the required features. The next version of your program might contain only one additional feature, and so on. Consider coding the simplest meaningful feature first, then coding the next simplest feature, and so on until your program is complete.
* Save versions of your program with different names. A working version with fewer features is more valuable than a nonworking version with more features. Don’t overwrite a working version with a nonworking version even if the nonworking version has more features.
* Review the scoring sheet (attached), assigning scores to your solution. Provide meaningful features to maximize the number of points that you earn. If there is a significant disparity between the score you assign to yourself and the score the TA determines, he or she will review the related features of your solution more carefully.
* The TA’s will look through your code even if things are not working perfectly, trying to award you as many points as possible. Include helpful comments in your code, and comment out sections of the code that aren’t working, but you feel are close to complete. Guide us with appropriate comments and we will do all we can to maximize your score.

**Midterm 1 Scoring Sheet**

Name:\_\_\_\_\_\_\_\_\_\_Section #:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Day Submitted: \_\_\_\_\_\_\_\_\_\_\_\_ T.A. \_\_\_\_

Days Late: \_\_\_\_\_ x 20= \_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Caution**: You will forfeit 10 points if your code has syntax errors and does not compile.

Student Grading TA Grading

**\_\_\_/ 55 pts \_\_\_/ 55 pts – Correct Algorithm**

\_\_\_/ 5 pts \_\_\_/5 pts –Accepts only positive numbers for dimensions (without crashing)

\_\_\_/ 15 pts \_\_\_/ 15 pts –Correctly determines tunnel dimensions and orientation

\_\_\_/ 15 pts \_\_\_/ 15 pts – Correctly determines the number of tunnels needed

\_\_\_/ 10 pts \_\_\_/10 pts – Correctly calculates and displays the covered area in square feet

\_\_\_/ 10 pts \_\_\_/10 pts – Correctly calculates and displays the covered percentage of the garden

plot

**\_\_\_/ 25 pts \_\_\_/ 25 pts – A Well Organized Program**

\_\_\_/ 10 pts \_\_\_/ 10 pts –The program runs as many times as the user desires

\_\_\_/ 15 pts \_\_\_/ 15 pts – Coherent output including all of the required prompts and results

**\_\_\_/ 20 pts \_\_\_/ 20 pts – Coding Style**

\_\_\_/ 10 pts \_\_\_/ 10 pts – Correct indentation, visual neatness

\_\_\_/ 10 pts \_\_\_/ 10 pts – Descriptive variable names, comments as needed

**\_\_\_/ 100 pts \_\_\_/ 100 pts – subtotal** (before late penalties)

**\_\_\_/ 100 pts \_\_\_/ 100 pts – subtotal** (after late penalties)

\_\_\_ Instruction #6 modification

\_\_\_ **TOTAL**

Student to TA Comments:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

TA to Student Comments:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_