



The sides of the cone are formed with the slash and back-slash characters respectively. The base of the cone is formed with the minus symbol. And the trunk is formed by the vertical bar or pipe symbol (the shift-backslash). There are two important and related concepts to keep in mind when writing output to the console:

1. Characters on each line are written left to right; once a character is written, it is not possible to backup and write a new character to the left of the last character written
2. Each line is written from the top of the console downward; once you begin a new line, it is not possible to backup and add a new line or change an existing line; each new line begins at the left edge of the console window

Program Requirements

Draw a pine tree as illustrated in the figure below. Specifically:

1. Name the file `tree.cpp`
2. Prompt the user for the height of the cone of the tree
3. Input the height of the cone from the console into a variable

4. If the cone height is less than 3 or greater than 15 (3 is okay and 15 is okay), then the program prints an error message and terminates (it does not loop or ask for new input)
5. Draw the cone part of the tree - the height of the cone is the height entered above
 - a. **No space between the left side of the console window and the left side of the cone at the base of the cone**
 - b. **No space between the sides of the cone at the peak**
6. Draw the base that separates the cone and the trunk with hyphen or minus characters not the underscore character
 - a. There will be an even number of dashes
 - b. **There is NO blank line between the cone and base**
 - c. **There is NO blank line between the base and the trunk**
7. Draw the tree trunk
 - a. **There are NO spaces between the sides of the trunk**
 - b. The trunk is half the height of the cone - if the height is odd, truncate (this is easy if you remember how [integer division](http://icarus.cs.weber.edu/~dab/cs1410/textbook/2.Core/common.html#division) (<http://icarus.cs.weber.edu/~dab/cs1410/textbook/2.Core/common.html#division>) works) - e.g., if the cone height is 5, the truck height is 2
8. Use only `iostream` and `iomanip` functions for I/O and formatting (no `stdio`)

Program Hints

1. See [Practice Problems 7 and 8](http://icarus.cs.weber.edu/~dab/cs1410/textbook/3.Control/practice.html) (<http://icarus.cs.weber.edu/~dab/cs1410/textbook/3.Control/practice.html>) and [pyramid.cpp](http://icarus.cs.weber.edu/~dab/cs1410/textbook/3.Control/progexample/pyramid.html) (<http://icarus.cs.weber.edu/~dab/cs1410/textbook/3.Control/progexample/pyramid.html>) (see especially the box highlighted in yellow)
2. For each character that forms the tree, ask yourself, "How many of these do I need?" For example, look at one row of the code
 - a. There are potentially many spaces needed between the left edge of the console window and the left edge of the tree - printing the spaces must be done in a loop
 - b. Similarly, there are potentially many spaces needed between the left and right sides of the cone - printing these spaces must also be done in a loop
 - c. Only one each of the `'/'` and `'\'` characters are needed per row - printing these characters is not done in a *nested* loop
 - d. It takes many rows to complete the cone - each row must be printed in a loop - an outer loop (see step 3)
3. **Use three for-loops to draw the cone:** two loops nested inside an outer loop:
 - a. The outer for-loop moves the cursor from the top of the cone to the bottom
 - b. The first nested for-loop to moves the cursor from the left edge of the screen to the left side of the cone
 - c. The second nested for loop to moves the cursor from the left side of the cone to the right side of the cone:

- d. See the table of [escape sequences](http://icarus.cs.weber.edu/~dab/cs1410/textbook/1.Basics/io.html#escape) [_\(http://icarus.cs.weber.edu/~dab/cs1410/textbook/1.Basics/io.html#escape\)](http://icarus.cs.weber.edu/~dab/cs1410/textbook/1.Basics/io.html#escape) in chap 2 to see how to draw a back-slash character
 - e. The statements that print the sides of the cone are outside of the nested loops
4. **Use one for-loop to draw the base** of the cone
5. **Use two for-loops, one nested inside the other, to draw the trunk**
- a. The outer loop moves the cursor from the top of the trunk to the bottom
 - b. The nested loop moves the cursor from the left edge of the screen to the left side of the trunk

Program Submission and Grading

- The program should exhibit good programming style.
- Upload `tree.cpp` to Canvas for grading.
- Be sure the program file is named correctly and that it does not include unspecified prompts, pauses, dummy reads, or menus. Make sure that the uploaded file is the correct file (i.e., verify that the contents are correct).

Lab 3 Scoring

Criteria	Ratings	Pts
<p>Requirement 1: tree.cpp</p> <p>The file name is correct (Canvas may add a "-1" or "-2" to the name, which is okay).</p>		1.0 pts
<p>Requirement 4: Boundary Conditions</p> <p>Trees that are too short or too tall are not drawn, and an appropriate message is printed.</p>		1.0 pts
<p>Requirement 5: Cone</p> <p>The cone is the correct height, the correct width, is drawn with the / and \ characters, and does not contain any unspecified characters.</p> <p>(a) No space between the left side of the console and the tree at the base of the cone.</p> <p>(b) No space between the sides of the cone at the peak.</p>		6.0 pts
<p>Requirement 6: Base</p> <p>The base is the correct length and is drawn with the minus character. (a) An even number of dashes. (b) No blank lines between the cone and the base. (c) No blank line between the base and the trunk. (d) No space between the left side of the console and the base.</p>		4.0 pts
<p>Requirement 7: Trunk</p> <p>The tree trunk is centered horizontally with respect to the cone and the base as illustrated.</p> <p>(a) No spaces between the sides of the trunk. (b) The tree trunk is the correct height (1/2 the cone height); odd cone heights produce the correct trunk height by truncating rather than rounding.</p>		5.0 pts
<p>Input</p> <p>Requirements 2 and 3: Prompt for the cone height and read the height from cin.</p>		1.0 pts
<p>Header Files</p> <p>#include directives are correct (stdfx.h and pch.h are removed or commented out).</p>		1.0 pts
<p>Miscellaneous</p> <p>The program is clean and easy to read: variable names are appropriate, indentation is consistent, blank lines are used appropriately; the program is created "Empty," and all "pause" statements and/or dummy reads are commented out or removed, etc. stdio not used.</p>		1.0 pts
Total Points: 20.0		

