Week 2: Discussion

By: Clinton E. Kline  
10-19-2021

Instructor: Quentin Miller

CIAT – ASD102

Abstract

Describe a program you are interested in writing with C++. After providing your description, provide the pseudo code for the main part of your program.

**Keywords**: C++, Programming, pseudo-code  
  
Week 2: Discussion

A program I would be interested in creating is an application that calculates the DIN setting, or binding retention setting of a ski binding for a given person.

The first step would be to take input to use as our variables. We would need to know the height and weight of the skier, the length of the sole of their ski boot, as well as their skill level on a scale of 1-3. 1 being a beginner, 3 being an expert.

The next step is going to be to create an array for each variable to be selected from. Our first set of array’s will be the boot sole arrays, a total of 6 arrays that will make up the body of our “DIN chart”. The elements of each bootSole array will be our DIN values ranging from smallest(loosest) to highest(tightest). The ranges of boot sole lengths will be sourced from a common DIN adjustment chart, which is generally standard throughout the industry. The next array will be the skierWeight array, which will include weight ranges in roughly 10lb increments. Last is the skierHeight array, similar to the weight array. The variable for the riders skill level can be input using an integer.

To begin our calculation we reference both the weight and height values that are input by the skier. Each will have an index value within their corresponding arrays. Whichever index value is lowest is chosen to be our reference value. This is because on a typical DIN chart, between the weight and height variables, you would use the value highest up on the chart(the value with the lowest index value) as your reference variable to compare to the variable referencing the skiers boot sole. If the skier is 175lbs and 5’9” tall we will be using the height variable as our reference. This is because on our DIN chart, the index value of the height range containing 5’9” is smaller than the index value of the weight range containing 175lbs. Once we have this value we simply reference the index value of the bootSole array that corresponds to the index value of the height array, in this example, to find our initial DIN value. We are not done yet however. Lastly, we will bring the skill level variable into the calculation. If the skiers skill level is 1, we use the value we already found. If the skill level is 2, we have to move to the next highest bootSole array, and if the skill level is 3 we move up 2 moot sole length arrays to ensure the nidings will be tighter fitting on skiers who are more aggressive.

So with that said, the pseudo-code for this application would look something like this:

var weight = user input;

var height = user input;

bootSole = user input;

skill level = 1 || 2 || 3;

weightArray[] = {a range of weights}

heightArray[] = {a range of heights}

#bootSole Arrays

lt250mm[] = {an array of din values}

251-270mm[] = {an array of din values}

271-290mm[] = {an array of din values}

291-310mm[] = {an array of din values}

311-330mm[] = {an array of din values}

gt331mm[] = {an array of din values}

main function {

switch statement:

in the case the bootsole is short,

use the corresponding array

in the case the bootsole is longer,

use the corresponding array

in the case the bootsole is even longer,

use the corresponding array

in the case the bootsole is even longer,

use the corresponding array, etc, etc..

return which array we're using

weightOrHeight() function; #refers to function below)

use the index value returned by the weightOrHeight function to indicate which DIN element to reference from our bootSole array;

return that value from the bootsole array;

skillLevel() function;

print the output!!

}

weightOrHeight function {

if the skiers weight is on the chart:

return the index value of the element which contains it;

if the skiers height is on the chart:

retirn the index value of the element which contains it;

if weight index is lower than height index:

use weight index value;

else

use height index value;

}

skillLevel function() {

switch statement:

in the case the skill level is 1:

din stays the same;

in the case the skill level is 2:

din is chosen from the array representing the next largest boot sole array

in the case the skill level is 3:

din is chosen from the array representing the boot size array that is two sizes larger.

return the new value to the main() function

}