Lab 2: Painting Class

Sam McDowell

9/15/2023

## Algorithm

1. Initialize Constants.
   1. This includes the number of gallons of paint used per square foot of wall.
2. Initialize Variables.
   1. These include a Room, the length, width, and height from the user, and two flow variables, shouldExit and isValid for the programs response to user input.
3. Start a loop that will continue until the user inputs “-1”.
   1. During the loop, the first step is to get the dimensions from the user.
      1. Ask the user for the length, width, and height of the room to be painted.
      2. Error check these dimensions to ensure they are in number format.
   2. Next the Room is updated with the stored values.
      1. Room does error checking when the values are input so ensure each value is updated successfully.
   3. If the inputs were invalid, prompt the user to enter new numbers.
      1. Flow returns to step 3a.
   4. If the inputs were valid, show a table of the data.
      1. The first part of this table is generated by Room.
         1. It includes the dimensions, volume and paintable surface area.
         2. Additionally, show the number of gallons that will be used when painting the room.
   5. Assuming the user did not enter “-1” flow will continue back at step 3a.

## Room

* (Constructor)
  + Set all dimensions to 0.
* Set Length
  + Update the length and return a Boolean representing if the input was valid.
* Set Width
  + Update the width and return a Boolean representing if the input was valid.
* Set Height
  + Update the height and return a Boolean representing if the input was valid.
* Get Length
  + Output the length of the room.
* Get Width
  + Output the width of the room.
* Get Height
  + Output the height of the room.
* Calculate Volume
  + Calculate the volume of the room.
  + Use the formula L\*W\*H.
* Calculate Paintable Area
  + Calculate the paintable surface area in the room.
  + Use the formula 2(L\*H) + 2(W\*H).
* Show Data
  + Print a table that shows the dimensions, the volume and the paintable area.

## Screenshots

A screenshot of a computer

Description automatically generated

Screenshot showing valid entry, then both types of error and re-prompt, then program exit.

## Integrity Statements

* I have not shared the source code in my program with anyone other than the pre-approved human sources.
* I have not used source code obtained from another student, or any other unauthorized source, either modified or unmodified.
* If any source code or documentation used in my program was obtained from another source, such as the course textbook or course notes, that has been clearly noted with a proper citation in the comments of my program.
* I have not knowingly designed this program in such a way as to defeat or interfere with the normal operation of any machine it is graded on or to produce apparently correct results when in fact it does not.