Assessment: Assignment01

Student Name: Yanzhang Wu

Lab Professor Name: *Fedor Ilitchev*

Lab Section Number: *CST8116\_323*

Due Date: June 17th 2022

# Understand the problem

This program will print several lines:

1. The input diameter
2. The number of coats needed
3. Numbers of tables that one can of varnish can paint
4. My name

A screenshot of a computer

Description automatically generated with medium confidence

Figure 1. Output of the console

The input of the program is the diameter of the table top and the number of coats needed. The area of the table top could be derived from the equation

where is the area, and is the diameter.

The total area can be calculated from

where n is number of coats needed.

Assume 1 can of varnish could coat area, then 1 can of varnish could paint tables where

For example, the diameter of the table top is 30 inches and we need 5 coats. Assuming and , we have

Therefore, for one can of varnish, it can paint 13.25 table tops (rounding is not required in this problem).

I will create three classes. The first one is class TableTop, in which the diameter will be the property and the area of one table top will be calculate in this class. The second class is class Varnish. This class will calculate the total area needed and numbers of tables that one can of varnish can paint. The third class is class Assignment1. By calling methods from the first 2 classes, I will use class Assignment1 to output information and test worker methods.

# Develop and Describe an Algorithm

## UML Class Diagrams

Text, letter

Description automatically generated

Figure 2. UML diagram of the program

## Pseudocode for class Assignment1

Start

//Declarations

num diameter

num coatsNeeded

num singleArea

num totalArea

num tablesPerOneCan

final num VARNISH\_COVERAGE

ouput "Please enter diameter (inches):"

enter diameter

if diameter is a positive number

break

else

output "Invalid input." and go back to output "Please enter diameter (inches):"

output "Please enter coats needed:"

enter coatsNeeded

if coatsNeeded is a positive integer

break

else

output "Invalid input." and go back to output "Please enter coats needed:"

//create top object

TableTop top

top.setDiameter(diameter)

singleArea = top.getSingleArea()

// create varnish object

varnish.setCoatsNeeded(coatsNeeded)

varnish.setVarnishCoverage(VARNISH\_COVERAGE)

totalArea = varnish.getTotalArea(singleArea)

tablesPerOneCan = varnish.getTablesPerOneCan(totalArea)

// output tablesPerOneCan and check results

output "Tables per one can: "+tablesPerOneCan

//ouput my name

output "Program by Yanzhang Wu"

End

## Pseudocode for class TableTop

TableTop()

//Declaration

num diameter

// non-argu constructor

public TableTop()

// constructor with arguments

TableTop(diameter)

this.diameter = diameter

//setter of diameter

setDiameter(diameter)

this.diameter = diameter

//getter of diameter

getDiameter()

return diameter

//area of one table top

getSingleArea()

//PI and Math.pow(num,num) are from Java API

num singleArea = PI \* Math.pow(diameter, 2) / 4

return singleArea

## Pseudocode for class Varnish

Varnish()

// declaration

num coatsNeeded;

num varnishCoverage;

// non-argu constructor

public Varnish()

// constructor with arguments

Varnish(coatsNeeded, varnishCoverage)

this.coatsNeeded = coatsNeeded

this.varnishCoverage = varnishCoverage

//setter of coatsNeeded

setCoatsNeeded(coatsNeeded)

this.coatsNeeded = coatsNeeded

//getter of coatsNeeded

getCoatsNeeded()

return coatsNeeded

//setter of varnishCoverage

setVarnishCoverage(varnishCoverage)

this.varnishCoverage=varnishCoverage

//getter of varnishCoverage

getVarnishCoverage()

return varnishCoverage

// calculate total area needed

getTotalArea(num singleArea)

num totalArea = singleArea \* coatsNeeded

return totalArea

// calculate tables per one can

getTablesPerOneCan (num totalArea)

num tablesPerOneCan = varnishCoverage/totalArea

return tablesPerOneCan

## Flowchart for class Assignment1

Diagram

Description automatically generated

Figure 3. Flowchart for class assignment1

## Flowchart for class TableTop

Diagram

Description automatically generated

Figure 4. Flowchart for class TableTop

## Flowchart for class Varnish

Diagram

Description automatically generated

Figure 5. Flowchart for class Varnish

# Algorithm Test plan

|  |  |  |  |
| --- | --- | --- | --- |
| input | Expected output | Actual output | Description |
| 42.5  4 | Please enter diameter (inches):  42.5  Please enter coats needed:  4  Tables per one can: 8.25  Program by Yanzhang Wu | Please enter diameter (inches):  42.5  Please enter coats needed:  4  Tables per one can: 8.25  Program by Yanzhang Wu | Matches (used a calculator) |
| 30  6 | Please enter diameter (inches):  30  Please enter coats needed:  6  Tables per one can: 11.04  Program by Yanzhang Wu | Please enter diameter (inches):  30  Please enter coats needed:  6  Tables per one can: 11.04  Program by Yanzhang Wu | Matches (used a calculator) |

# Source code

Graphical user interface, text, application

Description automatically generated

Figure 6. Source code of class Assignment1 - part 1

Text

Description automatically generated

Figure 7. Source code of class Assignment1 - part 2

Text

Description automatically generated

Figure 8. Source code of class TableTop

Text

Description automatically generated

Figure 9. Source code of class Varnish - part 1

Graphical user interface, text

Description automatically generated

Figure 10. Source code of class Varnish - part 2

# Screenshot of running program

Graphical user interface, text, application

Description automatically generated

Figure 11. Console output of running program

# Test Plan for the Program

|  |  |  |  |
| --- | --- | --- | --- |
| input | Expected output | Actual output | Description |
| 42.5  4 | Please enter diameter (inches):  42.5  Please enter coats needed:  4  Tables per one can: 8.25  Program by Yanzhang Wu | Please enter diameter (inches):  42.5  Please enter coats needed:  4  Tables per one can: 8.24742016520493  Program by Yanzhang Wu | Similar. Java has more decimals and is more accurate |
| 30  6 | Please enter diameter (inches):  30  Please enter coats needed:  6  Tables per one can: 11.04  Program by Yanzhang Wu | Please enter diameter (inches):  30  Please enter coats needed:  6  Tables per one can: 11.034742721038077  Program by Yanzhang Wu | Similar. Java has more decimals and is more accurate |
| asd  -5  0  (for diameter) | Invalid input.  Please enter diameter (inches): | Invalid input.  Please enter diameter (inches): | The actual output is identical to the expected output  (snapshot in next page) |
| qwe  -3  0  3.1  (for coats needed) | Invalid input.  Please enter coats needed: | Invalid input.  Please enter coats needed: | The actual output is identical to the expected output  (snapshot in next page) |

Text

Description automatically generated

Figure 12.Input "asd" for diameter

Text

Description automatically generated

Figure 13. Input "-5" for diameter

Text

Description automatically generated

Figure 14. Input "0" for diameter

A screenshot of a computer

Description automatically generated with medium confidence

Figure 15. Input "-3" for coats needed

Text, letter

Description automatically generated

Figure 16. Input "qwe" for coats needed

Text

Description automatically generated

Figure 17. Input "0" for coats needed

Text

Description automatically generated

Figure 18. Input "3.1" for coats needed