**CS1073  
FR03B**

**Assignment #4**

**Daniyal Khan  
3765942**

**Question I:**

**Source Code:**  
  
/\*\*

This class represents a cylindrical cup that can hold water

@author Daniyal Khan 3765942

\*/

public class CylinderCup {

/\*\*

Radius of the cylindrical cup

\*/

private double radius;

/\*\*

Height of the cylindrical cup

\*/

private double height;

/\*\*

Constructs a cylindrical cup given the radius and height

@param radius Radius of the cup

@param height Height of the cup

\*/

public CylinderCup(double radius, double height) {

this.radius = radius;

this.height = height;

}

/\*\*

Returns the radius of the cylindrical cup

@return Radius of the cup

\*/

public double getRadius() {

return radius;

}

/\*\*

Returns the height of the cylindrical cup

@return Height of the cup

\*/

public double getHeight() {

return height;

}

/\*\*

Returns the material required for making the cup which is the surface area

@return Surface area of the cup

\*/

public double materialRequired() {

return 2 \* Math.PI \* radius \* (radius + height) - Math.PI \* Math.pow(radius, 2);

}

/\*\*

Returns the amount of water cup can hold which is the volume

@return Volume of the cup

\*/

public double waterHoldAmount() {

return Math.PI \* Math.pow(radius, 2) \* height;

}

}

/\*\*

This class represents a conical cup that can hold water

@author Daniyal Khan 3765942

\*/

public class ConeCup {

/\*\*

Radius of the Conical Cup

\*/

private double radius;

/\*\*

Height of the Conical Cup

\*/

private double height;

/\*\*

Slant Height of the Conical Cup

\*/

private double slantHeight;

/\*\*

Constructs a Conical Cup given the radius and height

@param radius Radius of the Cup

@param height Height of the Cup

\*/

public ConeCup(double radius, double height) {

this.radius = radius;

this.height = height;

slantHeight = Math.sqrt((Math.pow(radius, 2)) + (Math.pow(height, 2)));

}

/\*\*

Returns the radius of the Conical Cup

@return radius of the cup

\*/

public double getRadius() {

return radius;

}

/\*\*

Returns the height of the Conical Cup

@return height of the cup

\*/

public double getHeight() {

return height;

}

/\*\*

Returns the slantHeight of the Conical Cup

@return slantHeight of the cup

\*/

public double slantHeight() {

return slantHeight;

}

/\*\*

Returns the material required for building the concical cup which is the surface area

@return Surface area of the Cup

\*/

public double materialRequired() {

return Math.PI \* radius \* (radius + slantHeight) - Math.PI \* Math.pow(radius, 2);

}

/\*\*

Returns the amount of water cup can hold which is the volume

@return Volume of the Cup

\*/

public double waterHoldAmount() {

return (double)1/3 \* Math.PI \* Math.pow(radius, 2) \* height;

}

}  
  
  
  
**Driver:**

import java.util.Scanner;

import java.text.NumberFormat;

/\*\*

This is a driver class which uses the CylinderCup and ConeCup classes

@author Daniyal Khan 3765942

\*/

public class Driver {

public static void main (String[] args) {

Scanner scan = new Scanner(System.in);

NumberFormat format = NumberFormat.getNumberInstance();

int input = 0;

double largestSurfaceAreaCone = 0.0;

double largestSurfaceAreaCylinder = 0.0;

format.setMaximumFractionDigits(3);

format.setMinimumFractionDigits(3);

while (input != 3) {

System.out.print("\nSelect one of the following options: \n");

System.out.print("1 - Enter the information for a conical cup\n");

System.out.print("2 - Enter the information for a cylindrical cup\n");

System.out.print("3 - Quit\n");

System.out.print("Enter you choice: ");

input = scan.nextInt();

scan.nextLine(); // consumes the nextline character

switch (input) {

case 1:

System.out.print("Length of radius (cm): ");

double radiusConical = scan.nextDouble();

scan.nextLine(); // consumes nextline character

System.out.print("Length of height (cm): ");

double heightConical = scan.nextDouble();

scan.nextLine(); // consumes nextline character

ConeCup conical = new ConeCup(radiusConical, heightConical);

double surfaceAreaConical = conical.materialRequired();

double volumeConical = conical.waterHoldAmount();

System.out.print("Surface area: " + format.format(surfaceAreaConical) + "cm^2");

System.out.println("\nVolume: " + format.format(volumeConical) + "cm^3");

if (surfaceAreaConical > largestSurfaceAreaCone) {

largestSurfaceAreaCone = surfaceAreaConical;

}

largestSurfaceAreaCone = surfaceAreaConical;

break;

case 2:

System.out.print("Length of radius (cm): ");

double radiusCylinder = scan.nextDouble();

scan.nextLine(); // consumes the nextline character

System.out.print("Length of height (cm): ");

double heightCylinder = scan.nextDouble();

scan.nextLine(); // consumes nextline character

CylinderCup cylinderical = new CylinderCup(radiusCylinder, heightCylinder);

double surfaceAreaCylinder = cylinderical.materialRequired();

double volumeCylinder = cylinderical.waterHoldAmount();

System.out.print("Surface area: " + format.format(surfaceAreaCylinder) + "cm^2");

System.out.println("\nVolume: " + format.format(volumeCylinder) + "cm^3");

if (surfaceAreaCylinder > largestSurfaceAreaCylinder) {

largestSurfaceAreaCylinder = surfaceAreaCylinder;

}

largestSurfaceAreaCylinder = surfaceAreaCylinder;

break;

case 3:

break;

default:

System.out.println("Not an available option");

}

}

System.out.print("\nThe cup with the largest surface area is a " + (largestSurfaceAreaCone > largestSurfaceAreaCylinder? "Cone" : "Cylinder"));

System.out.println("\nThe surface area is: " +

format.format(largestSurfaceAreaCone > largestSurfaceAreaCylinder? largestSurfaceAreaCone : largestSurfaceAreaCylinder) + "cm^2");

}

}

**Output:**

A screenshot of a computer program

Description automatically generated