

Answers

1. Volume: 331.6625 cm^3 ; label area: 204.1 cm^2 ;
design cost: \$306.15
2. a) The jars have similar volumes. To the nearest cubic centimetre, the volume of the first jar is 593 cm^3 and the volume of the second jar is 580 cm^3 .
b) The first jar; to the nearest cent, the label design cost for the first jar is \$256.22 and the label design cost for the second jar is \$296.26.
3. In the main program, I would set the **pricePerSquareCM** variable to 1.25 instead of 1.5.
4. Sample answer:
The first part of the subprogram **collectInfo** would change to

```
ask for the container radius
```


In the subprogram **calculateVolume**, I would delete the line that determines the radius.
I would change the subprogram **calculateLabelArea**, like this:

```
subprogram calculateLabelArea  
    labelArea = 2 * pi * radius * labelHeight  
display labelArea
```

Answers (cont'd)

5. Sample answer:

- a) I would create a new variable for the surface area of the top, called **topSA**.

In the pseudocode, I would add a new subprogram called **calculateTopSA**:

```
subprogram calculateTopSA
  topSA = pi * radius * radius
  display topSA
```

In the program, it would look like this:

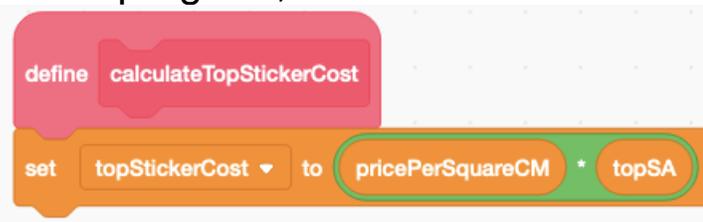


- b) I would create a new variable for the cost to design the sticker, called **topStickerCost**.

In the pseudocode, I would add a new subprogram called **calculateTopStickerCost**:

```
subprogram calculateTopStickerCost
  topStickerCost = pricePerSquareCM * topSA
  display topStickerCost
```

In the program, it would look like this:

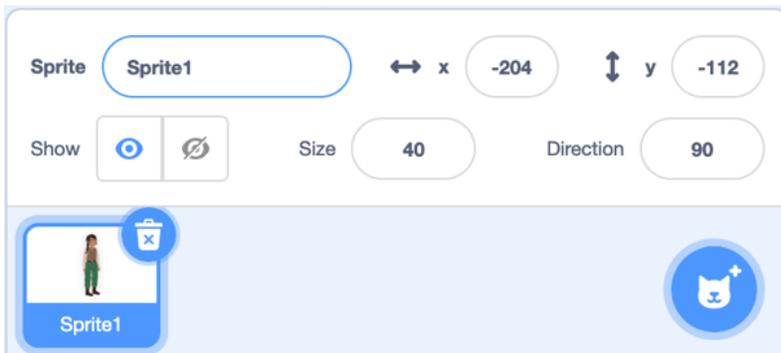


Answers (cont'd)

Extensions:

Sample answers:

- Below the stage, I selected **Show** to have the sprite appear, made the size 40, and dragged the sprite to the lower left corner of the stage.



Then, at the end of the main program, I added a block that made the sprite announce what the design cost was.



- I added a block to the end of the label design cost subprogram. The new block multiplies the **designCost** variable by 100, rounds it, then divides by 100.

The block looks like this:



- Samples using grocery items will vary.

Answers (cont'd)

- Sample program based on modifying Yindi's program. For this sample, an image was added to help clients visualize the box and which face will be considered the base.

```

when clicked
  set pricePerSquareCM to 1.5
  collectInfo
  calculateVolume
  calculateLabelArea
  calculateDesignCost

define collectInfo
  ask "What's the length of the box?" and wait
  set length to answer
  ask "What's the width of the box?" and wait
  set width to answer
  ask "What's the height of the box?" and wait
  set height to answer

define calculateVolume
  set volume to length * width * height

define calculateLabelArea
  set labelArea to 2 * length * height + 2 * width * height

define calculateDesignCost
  set designCost to pricePerSquareCM * labelArea
  
```

Output for box with length 6 cm, width 4 cm, and height 8.5 cm.

