Geometric warehouse

Imagine we are building a warehouse that will house a number of geometric shapes. It can store up to 100 shapes at any time. It can identify each of the shapes individually. It can also tell use what the total area all the shapes are occupying at any given time.

On the final version of the application, it will accept only three types of shapes: circles, rectangles and cubes. The application will hold the minimal information for each shape as follows:

Circle	Rectangle	Cube
Shape id: integer Diameter: float	Shape id: integer Height: float Width: float	Shape id: integer Height: float Width: float Depth: float

The program will have a menu with the following options:

1. **Add a circle**
2. **Add a rectangle**
3. **Add a cube**
4. **List items**
5. **Get Statistic**
6. **Exit**

The activities of each of the menu's item will be following

1. Add a circle
Once selected, it will ask the user to type the followings:

What is the diameter:

Once the user type in one value, it will assign a unique integer id number (Starting with 1) to this shape and save the circle diameter into memory (array)

2. Add a Rectangle
Once selected, it will ask the user to type the followings:

What is the height:

What is the width:

Once the user type in two values, it will assign a unique integer id number (continuation from the previous Add circle or Rectangle or cube) to this shape and save the rectangle parameters into memory (array)

3. Add a Cube
Once selected, it will ask the user to type the followings:

What is the height:
What is the width:
What is the depth:

Once the user type in three values, it will assign a unique integer id number (continuation from the previous Add circle or Rectangle or cube) to this shape and save the cube parameters into memory (array)

4. List Items:
When selected, it will show the items ordered by the id numbers as following example:

```
Id        type           dimension
==============================
1      circle                25
2      rectangle             41 x 37
3      rectangle             53 x 59
4      cube                  31 x 12 x 45
```

5. Get Statistic: (Given Sample output may not be correct)
When selected, it will show the following details:

Total shapes: 4
Total number of Rectangles: 2
Total number of Circles: 1
Total number of Cube: 1
Total area: 4995.68
The total area occupied by rectangles: 4644 (92.96%)

The total area occupied by circles: 150.68 (2.00%)
The total area occupied by circles: 301.68 (5.04%)

6. Exit

Use C programming and construct. Create separate programmes:
1. **Without Structure**
2. **With Structure**
3. **With nested structure**