

(a) Fill in the blanks below:

```

Class Main {
    public void main(String[] args){
        Planet kepler = new Exoplanet();
        Sphere jupiter = new Sphere();
        Sphere arion = new Exoplanet();
        Planet earth = new Planet();

        kepler.orbit((Exoplanet) arion);
        ((Exoplanet) earth).orbit(kepler);
        ((Exoplanet) kepler).getDistance();
        earth.getRadius();
        arion.getDistance();
        earth.radius;
        Exoplanet.getDistance();
    }
}

```

\_\_\_\_\_ 1  
 \_\_\_\_\_ 2  
 \_\_\_\_\_ 3  
 \_\_\_\_\_ 4  
 \_\_\_\_\_ 5  
 \_\_\_\_\_ 6  
 \_\_\_\_\_ 7  
 \_\_\_\_\_ 8  
 \_\_\_\_\_ 9  
 \_\_\_\_\_ 10  
 \_\_\_\_\_ 11

(b) Within the blank below, briefly explain why changing the access modifier on line 2 to, **private default void getRadius()**, would result in a compilation error. Answer in 10 words or less.

(c) Within the class `Exoplanet`, we define a new method below. Which of the following changes will make `Exoplanet` compile?

```
public void fixRadius() { super.radius = 6; }
```

- Change the `private` keyword to `public` on line 6.
- Modify `fixRadius()` above to be `private`.
- Modify `fixRadius()` above to be `static`.

(d) True or false: we can create another interface that extends `Sphere`.

- True
- False

(e) It is \_\_\_\_\_ possible to implement two interfaces in the same class.

- Always
- Sometimes
- Never