A (b) 4 ✓

A (c) 20, 27, 24, 26

A (d) 10

A (e) $3+3+3+2 = \boxed{11 \text{ comparisons}}$ ✓

B(a)

pass 1: 4 2 1 5 6 | 9 ✓
pass 2: 2 1 4 5 | 6 9 ✓
pass 3: 1 2 4 | 5 6 9 ✓
pass 4: 1 2 | 4 5 6 9 ✓

B(b) during pass 1: (6,4), (6,2), (6,1), (6,5), (6,9) ✓
during pass 2: (4,2), (4,1), (4,5), (5,6) ✓
during pass 3: (2,1), (2,4), (4,5) ✓
during pass 4: (1,2), (2,4) ✓

B(c) 4+2+1 = 7 swaps ✓

B(d) 4 passes are required to sort the array. ✓
(5+4+3+2)+(6+5+4+3)+4 =
14+18+4 =
36 number of comparisons in total. ✓

C [[2,1,3],[4,7,11],[18,29,47]] i ✓

D(a)

```
public class Cylinder
{
    private double height;
    private double radius;

    public Cylinder()
    {
       height = 0;
       radius = 0;
    }

    public Cylinder(double h, double r) throws IllegalArgumentExceptio
    {
       if (h<0 || r<0)
       {
           throw new IllegalArgumentException("Invalid input.");
       }
       height = h;
       radius = r;
    }

    public double getHeight()
    {
       return height;
    }
```

D(a) cont.

```
  public double getRadius()   ✓
  {
      return radius;
  }

   public void setCylinder(double h,double r) throws IllegalArgumentException
   {
      if (h<0 || r<0)   ✓
      {
        throw new IllegalArgumentException("Invalid input.");
      }
      height = h;
      radius = r;
   }

  public String toString()   ✓
  {
     return "Cylinder[height = "+height+", radius = "+radius+"]";
  }
}
```

D(b)

```
Cylinder can = new Cylinder(6,4);
```

✓

D(c)

```
Systemout.printf("area = %f squared inches", 2*Math.PI*
        can.getRadius() * (can.getRadius()+ can.getHeight()));
```

✓

D(d)

```
can.setCylinder(can.getHeight(), 2* can.getRadius());
```

✓