Question 1.

```
/** = the value of the expression. */
public boolean eval() {
    if (kind.equals("true")) return true;
    if (kind.equals("false")) return false;
    if (kind.equals("&&"))
       return op1.eval() && op2.eval();
    if (kind.equals("||"))
       return op1.eval() || op2.eval();
    // kind = "()"
    return op1.eval();
}
```

Question 2. Process m..n.

```
int i= 0;
// inv:  b[m..i–1] contains the odd values in original b[m..k-1]
for (int k= 0; k <= n; k= k+1) {
   // Process k
   if (b[k]%2 == 1) {
      b[i]= b[k];  i= i+1;
   }
}
// b[m..i–1] contains the odd values in original array b[m..n]
```

Question 3.

```
k= v.size() – 1;
// inv:  v[k+1..] does not contain w
while (0 <= k  &&  ((w == null  && v.get(k) != null) ||
                    (w != null  &&  !w.equals(v.get(k))))) {
    k=  k – 1;
}
```

Question 4.

(a) and (b) See diagrams below

(c) true, true, true, false, false

(d) Uses of a wrapper class.

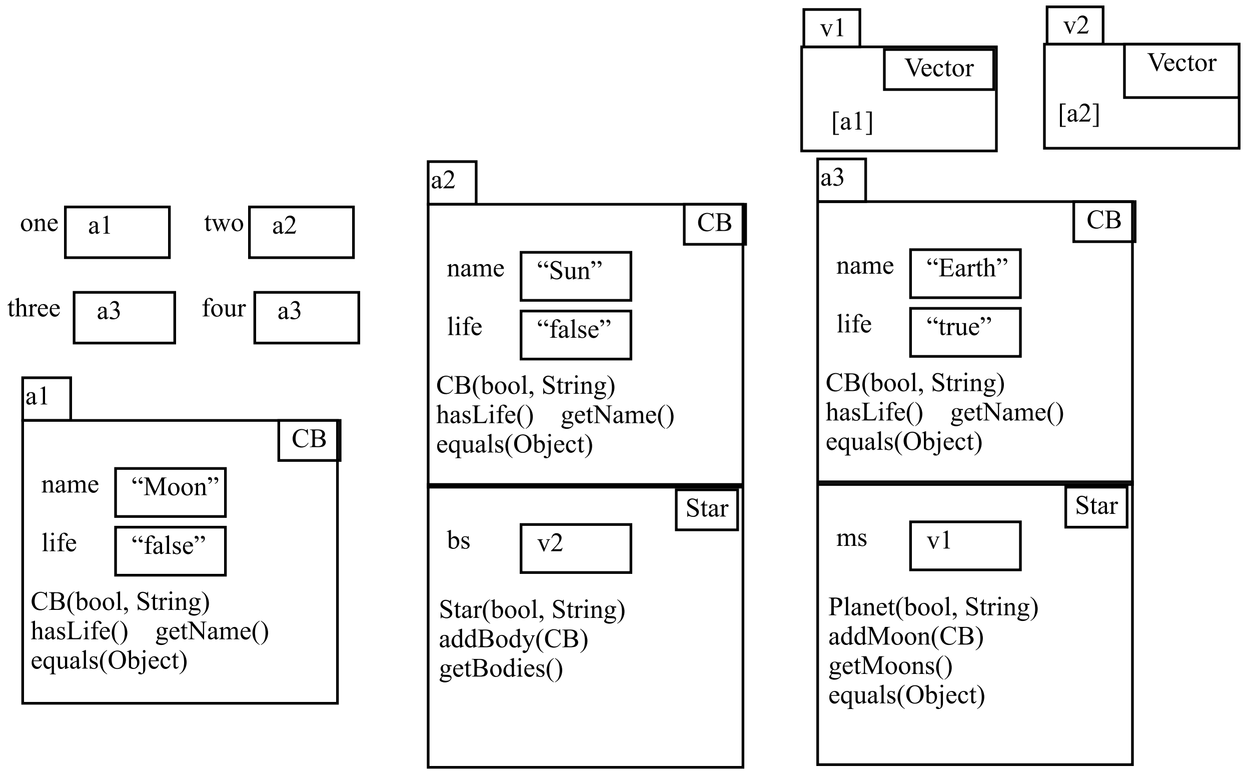
1. Wrap one value of a primitive type so that it can be treated as an object.
2. Hold useful method dealing with the type with which the wrapper class is associated ()e.g Integer with **int**).

(e)

```
/** b is a Planet and has the same name, life
    property, and moons as this Planet. */
public boolean equals(Object b) {
    if (!(b instanceof Planet))  return false;
    if (!super.equals(b))  return false;
    Planet bp= (Planet)b;

    // Return true if this and bp have same moons
    if (moons.size() != bp.moons.size())
        return false;

    // inv: moons[0..k-1] same as bp.moons[0..k-1]
    for (int k= 0; k < moons.size(); k= k+1) {
       if (!(moons.get(k).equals(bp.moons.get(k))))
          return false;
    }
    return true;
}
```



v1
Vector
[a1]
v2
Vector
[a2]
one a1
two a2
three a3
four a3
a1
CB
name "Moon"
life "false"
CB(bool, String)
hasLife() getName()
equals(Object)
a2
CB
name "Sun"
life "false"
CB(bool, String)
hasLife() getName()
equals(Object)
Star
bs v2
Star(bool, String)
addBody(CB)
getBodies()
a3
CB
name "Earth"
life "true"
CB(bool, String)
hasLife() getName()
equals(Object)
Star
ms v1
Planet(bool, String)
addMoon(CB)
getMoons()
equals(Object)