Timothy Hawkins

Assignment #4

```
@Test
public void PCTest1(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(63);
    thermo.setThresholdDiff(5);

    //clause b: Override
    thermo.setOverride(false);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(63);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertTrue(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void PCTest2(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(63);
    thermo.setThresholdDiff(5);

    //clause b: Override
    thermo.setOverride(true);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(75);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(12);
    thermo.setTimeSinceLastRun(10);

    assertFalse(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void PCTest3(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(65);
    thermo.setThresholdDiff(3);

    //clause b: Override
    thermo.setOverride(true);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(60);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertTrue(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void PCTest4(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(65);
    thermo.setThresholdDiff(3);

    //clause b: Override
    thermo.setOverride(false);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(60);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertTrue(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void CCTest1(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(63);
    thermo.setThresholdDiff(5);

    //clause b: Override
    thermo.setOverride(true);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(75);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertTrue(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void CCTest2(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(65);
    thermo.setThresholdDiff(9);

    //clause b: Override
    thermo.setOverride(false);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(63);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(12);
    thermo.setTimeSinceLastRun(10);

    assertFalse(thermo.turnHeaterOn(ps));
}
```

```java
@Test
public void CACCTest1(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(63);
    thermo.setThresholdDiff(5);

    //clause b: Override
    thermo.setOverride(true);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(75);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertTrue(thermo.turnHeaterOn(ps));
}
```

```java
@Test
public void CACCTest2(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(66);
    thermo.setThresholdDiff(9);

    //clause b: Override
    thermo.setOverride(true);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(75);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertFalse(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void CACCTest3(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(65);
    thermo.setThresholdDiff(5);

    //clause b: Override
    thermo.setOverride(true);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(75);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertTrue(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void CACCTest4(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(65);
    thermo.setThresholdDiff(5);

    //clause b: Override
    thermo.setOverride(false);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(75);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(10);
    thermo.setTimeSinceLastRun(12);

    assertFalse(thermo.turnHeaterOn(ps));
}
```

```
@Test
public void CACCTest5(){
    //clause a: curTemp < dTemp - thresholdDiff
    thermo.setCurrentTemp(63);
    thermo.setThresholdDiff(5);

    //clause b: Override
    thermo.setOverride(true);

    //clause c: curTemp < overTemp - thresholdDiff
    thermo.setOverTemp(70);

    //clause d: timeSinceLastRun > minLag
    thermo.setMinLag(12);
    thermo.setTimeSinceLastRun(10);

    assertFalse(thermo.turnHeaterOn(ps));
}
```