

Vejledende løsninger

Opgave 1

Spørgsmål 1.1

```
String toBinary(int d) {
    IntStack s = new IntStack(31);
    while (d > 0) {
        int r = d%2;
        s.push(r);
        d /= 2;
    }
    String result = "";
    while (!s.empty())
        result += s.pop();
    return result;
}
```

Spørgsmål 1.2

```
String toBinary(int d) {
    return d > 0 ? toBinary(d/2) + d%2 : "";
}
```

Opgave 2

Spørgsmål 2.1

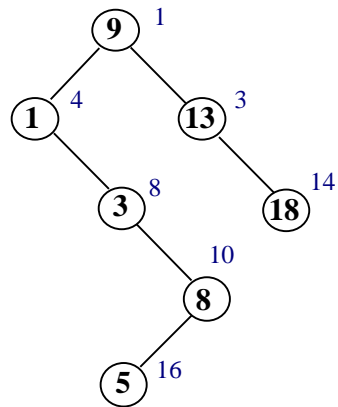
```
private Node find(Node n) {
    while (n.dad != null)
        n = n.dad;
    return n;
}
```

Spørgsmål 2.2

```
Node root1 = find(array[i]);
Node root2 = find(array[j]);
if (root1 != root2)
    root2.dad = root1;
```

Opgave 3

Spørgsmål 3.1



Spørgsmål 3.2

```
Comparable find(Comparable k) {
    TreapNode t = this;
    while (t != null) {
        if (k.compareTo(t.key) < 0)
            t = t.left;
        else if (k.compareTo(t.key) > 0)
            t = t.right;
        else
            return t.key;
    }
    return null;
}
```

Spørgsmål 3.3

```
TreapNode rotateWithLeftChild() {
    TreapNode c = left;
    left = c.right;
    c.right = this;
    return c;
}
```

Spørgsmål 3.4

A:

```
if (left.priority < priority)
    return rotateWithLeftChild();
```

B:

```
if (right.priority < priority)
    return rotateWithRightChild();
```

Spørgsmål 3.5

C:

```
return (left.priority < right.priority ?
        rotateWithLeftChild() :
        rotateWithRightChild()).remove(k);
```

Opgave 4

Spørgsmål 4.1

```
IntTree(int actualData, Tree actualLeft, Tree actualRight) {  
    super(actualLeft, actualRight);  
    data = actualData;  
}
```

Spørgsmål 4.2

```
void visit() {  
    if (isLeaf())  
        System.out.print(data);  
}
```

Spørgsmål 4.3

Ved at ombytte

```
    if (left != null) left.traverse();
```

med

```
    if (right != null) right.traverse();
```

Opgave 5

Spørgsmål 5.1

```
void update(int temperature) {  
    if (temperature < threshold)  
        on = true;  
    else  
        on = false;  
}
```

Spørgsmål 5.2

```
EconomyThermostate(int actualMaxThreshold) {  
    maxThreshold = actualMaxThreshold;  
}
```

Spørgsmål 5.3

```
void setThreshold(int actualThreshold) {  
    if (actualThreshold < maxThreshold)  
        super.setThreshold(actualThreshold);  
    else  
        super.setThreshold(maxThreshold);  
}
```

Opgave 6

Spørgsmål 6.1

```
Dato(int dag, int maaned, int aar) {  
    this.dag=dag;  
    this.maaned=maaned;  
    this.aar=aar;  
}
```

Spørgsmål 6.2

```
public String toString() {  
    return ""+dag+"/"+maaned+"-"+aar;  
}
```

Spørgsmål 6.3

```
boolean foer(Dato d) {  
    if(aar==d.aar&&maaned==d.maaned)  
        return dag<d.dag;  
    else  
        if(aar==d.aar)  
            return maaned<d.maaned;  
        else  
            return aar<d.aar;  
}
```

Spørgsmål 6.4

```
String status(Dato idag) {  
    if(idag.foer(start))  
        return "ej startet";  
    else  
        if(idag.foer(slut))  
            return "undervejs";  
        else  
            return "færdig";  
}
```

Spørgsmål 6.5

```
Projekt(String t,Dato d1,Dato d2) {
    titel=t;
    start=d1;
    slut=d2;
}

ProjektMedMilepael(String t,Dato d1,Dato d2, Dato d3) {
    super(t,d1,d3);
    milepael=d2;
}
```

Spørgsmål 6.6

```
String status(Dato idag) {
    if(idag.foer(start))
        return "ej startet";
    else
        if(idag.foer(milepael))
            return "før milepæl";
        else
            if(idag.foer(slut))
                return "efter milepæl";
            else
                return "færdig";
}
```

Spørgsmål 6.7

```
String status(Dato idag) throws Exception {
    if (start==null||milepael==null||slut==null)
        throw new Exception();
    if(slut.foer(milepael)||milepael.foer(start))
        throw new Exception();
    ...
}
```