

```
void delete (int value)
```

```
{ if (Nodes != null && (Nodes.findNodeWithKey(value) != null)
```

```
{ decreaseKey value (value, find minimum () - 1)
```

```
extractMin();
```

```
}
```

```
}
```

```
public decreaseKey value (int old value, int new-value)
```

```
{ BinomialHeapNode temp = Nodes.findNodeWithKey (old-value);
```

```
if (temp == null) return;
```

```
temp.key = new-value;
```

```
BinomialHeapNode tempParent = temp.parent;
```

```
while (tempParent != null && temp.key < tempParent.key)
```

```
{ int z = temp.key;
```

```
temp.key = tempParent.key;
```

```
tempParent.key = z;
```

```
temp = tempParent;
```

```
tempParent = tempParent.parent;
```

```
}
```

```
}
```