Figure 14.1 Superclass extended by subclasses.

Here is a class for letter grades:

```
public class LetterGrade extends Mark {
    private String mark;

    public LetterGrade (String mark) {
        this.mark = mark;
    }

    public int intValue () {
        if (mark.equals("A+")) return 95;
        if (mark.equals("A")) return 88;
        …
    }
}
```

and here is one for numerical grades:

```
public class PercentGrade extends Mark {
    private int mark;

    public PercentGrade (String mark) {
        this.mark = Integer.parseInt(mark);
    }

    public int intValue () {
        return mark;
    }
}
```

Both classes **LetterGrade** and **PercentGrade** *extend* class **Mark**.
Each is a “kind of” **Mark**, but there are subtle (or maybe not-so-subtle) differences between the two.

We also need a class **Mark** for these to be extensions of:

```
public class Mark {
    // why don't we need a constructor?

    public int intValue () {
        return -9999;    // this will be overridden
    }
}
```

We say that **LetterGrade** and **PercentGrade** are *child* classes or *subclasses*, and **Mark** is the *parent* class or *superclass*.