

The first example is of a function call stack (described in class). Here is some fictitious code for computing employee bonuses in a large and silly corporation.

```
def calculatebonus(salary, education, height):
    ratio = thousandspergrade(salary, education)
    ratio = ratio + 1
    return valuetimesstature(ratio, height)

def thousandspergrade(pay, grade):
    return (pay / 1000) / grade

def valuetimesstature(factor, feettall):
    metersaboveave = heightconverter(feettall)
    return factor * metersaboveave

def heightconverter(feet):
    return (feet - 5.75) / 3.28
```

Here is a recursive function to compute the factorial. How does the call stack behave, when we invoke `factorial(4)`?

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)
```

The second example comes from Extensible Markup Language (XML), which has become a popular data-description language throughout the computer industry. Well-formed XML code is “stacky”, and XML-processing programs are *required* by the XML specification to reject code that is not well-formed according to the stackiness idea. Here is some XML for describing a list of songs; in what way is it stacky?

```
<songlist>
  <song>
    <title>Help Me Mary</title>
    <composer>Liz Phair</composer>
    <producer>Liz Phair</producer>
    <producer>Brad Wood</producer>
```

```
<publisher>Matador</publisher>
<length>2:16</length>
<year>1993</year>
<artist>Liz Phair</artist>
</song>
<song>
  <title>Hot Cop</title>
  <composer>Jacques Morali</composer>
  <composer>Henri Belolo</composer>
  <composer>Victor Willis</composer>
  <producer>Jacques Morali</producer>
  <publisher>Polygram</publisher>
  <length>6:20</length>
  <year>1978</year>
  <artist>Village People</artist>
</song>
</songlist>
```

HTML is the traditional language for describing web pages. Officially, HTML is supposed to be stacky, like XML. Unofficially, many web browsers happily accept ill-formed HTML, such as the following snippet. In what way is it ill-formed? Can you rearrange the code to make it well-formed?

```
<html>
  <body>
    <blockquote>
      13<sup><i>x</sup></i> <b>v</b>
    </blockquote>
  </body>
</html>
```

Because many browsers accept ill-defined HTML, many web designers are sloppy about how they write HTML, and their web pages display correctly in some browsers but not in others. Such incompatibility inconveniences users and costs web designers time and money. One proposed remedy is to replace HTML with a new language, for which well-formed documents are strictly required. XHTML, a dialect of XML that produces functionality similar to HTML, is an emerging contender for this new language standard.