

A diagram illustrating recursive calls for the `binaryToDecimal` function. It features several colored circles (purple, grey, yellow, red) and arrows indicating the flow of the recursion. A purple circle is at the top, with a grey circle below it. A yellow circle is to the left of the `return 2 * binaryToDecimal(str, position - 1);` line, with an arrow pointing to the `position - 1` expression. Another yellow circle is to the left of the closing brace of the `if` block. A red circle is below the `return 1 + 2 * binaryToDecimal(str, position - 1);` line, with an arrow pointing to the `position - 1` expression. A final red circle is at the bottom left, below the closing brace of the function.

```
static int binaryToDecimal(String str, int position) {  
    if (position < 0) {  
        return 0;  
    } else if (str.charAt(position) == '0') {  
        return 2 * binaryToDecimal(str, position - 1);  
    }  
  
    return 1 + 2 * binaryToDecimal(str, position - 1);  
}
```