

The diagram illustrates the control flow of the provided Java code. It features several colored nodes (yellow, orange, and red) and arrows indicating the sequence of execution. A yellow node at the top left points to the start of the function. An orange node points to the 'if' statement. A red node points to the 'return 0;' statement. Another orange node points to the 'else if' statement. A yellow node points to the 'return 1 + 2 * ...' statement. Arrows show the flow from the 'if' statement to the 'return 0;' statement, and from the 'else if' statement to the 'return 1 + 2 * ...' statement. There is also a direct arrow from the 'if' statement to the 'return 1 + 2 * ...' statement, suggesting a jump or a specific flow path.

```
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);  
}
```