1. Draw a parse tree for the function below. You can assume that the “for” statement is at the top of the parse tree.

```c
void copy_bytes(char dest[], char source[], int n) {
    for (int i = 0; i < n; ++i)
        dest[i] = source[i];
}
```
2. Write pseudocode for a simple AST-walker analysis that warns when string concatenation occurs in a loop. You may assume Note:

- In Java and .NET it is more efficient to use a StringBuffer
- Assume any appropriate AST elements that you need

To get you started:

```java
class StringConcatLoopAnalysis extends Visitor {
    void visitStringConcat(StringConcat e) {
        // Pseudocode for analysis
    }
}
```