```java
import components.simplereader.SimpleReader;

/**
 * Program to evaluate XMLTree expressions of integers.
 *
 * @author Vishal Kumar
 *
 */
public final class XMLTreeIntExpressionEvaluator {

    /**
     * Private constructor so this utility class cannot be instantiated.
     */
    private XMLTreeIntExpressionEvaluator() {
    }

    /**
     * Evaluate the given expression.
     *
     * @param exp
     *            the {@code XMLTree} representing the expression
     * @return the value of the expression
     * @requires <pre>
     * [exp is a subtree of a well-formed XML arithmetic expression]  and
     *  [the label of the root of exp is not "expression"]
     * </pre>
     * @ensures evaluate = [the value of the expression]
     */
    private static int evaluate(XMLTree exp) {
        assert exp != null : "Violation of: exp is not null";
        int value = 0;

        // statement to get past the first root node
        if (exp.label().equals("expression")) {
            value += evaluate(exp.child(0));
        }
        // base case
        else if (exp.hasAttribute("value")) {
            value += Integer.parseInt(exp.attributeValue("value"));
        }
        // recursive if else block to evaluate math operations
        else if (exp.label().equals("plus")) {
            value += evaluate(exp.child(0)) + evaluate(exp.child(1));
        } else if (exp.label().equals("minus")) {
            value += evaluate(exp.child(0)) - evaluate(exp.child(1));
        } else if (exp.label().equals("times")) {
            value += evaluate(exp.child(0)) * evaluate(exp.child(1));
        } else {
            value += evaluate(exp.child(0)) / evaluate(exp.child(1));
        }

        // return final value
        return value;
    }

    /**
     * Main method.
     *
     * @param args
     *            the command line arguments
     */
    public static void main(String[] args) {
        SimpleReader in = new SimpleReader1L();
        SimpleWriter out = new SimpleWriter1L();

        out.print("Enter the name of an expression XML file: ");
        String file = in.nextLine();
        while (!file.equals("")) {
            XMLTree exp = new XMLTree1(file);
            out.println(evaluate(exp.child(0)));
            out.print("Enter the name of an expression XML file: ");
            file = in.nextLine();
        }

        in.close();
        out.close();
    }

}
```