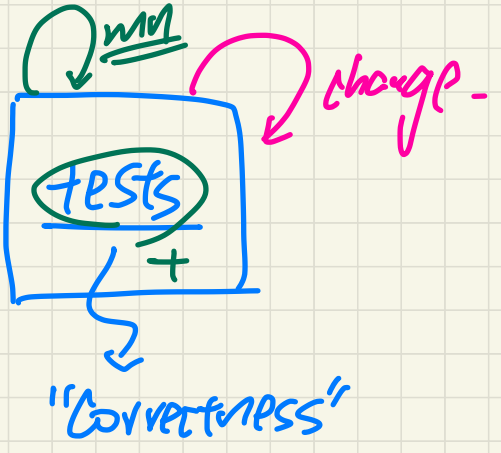
Lecture 14 - Nov. 1

Visitor, Syntactic Analysis

Visitor: Double Dispatch
Visitor: Open-Closed Principle
Visitor: Single-Choice Principle

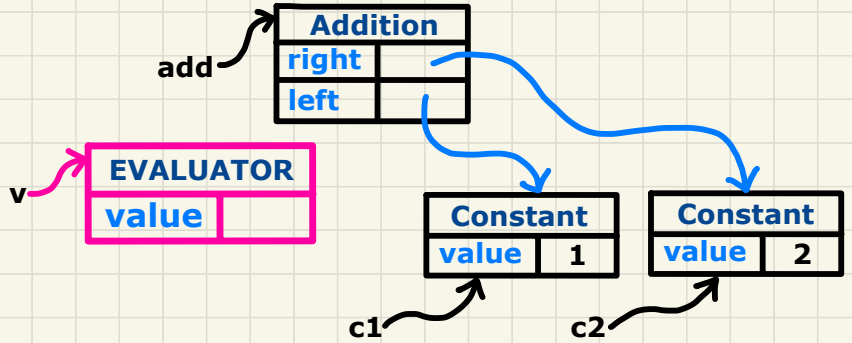
Announcements

- **Assignment 2** released
  + Python script for **Regression Testing**



- **Project Milestone 1** due next week
  + Sign-Up sheet activated tomorrow (Wednesday) at 6pm

Executing Composite and Visitor Patterns at Runtime



Tracing add.accept(v)
Double Dispatch

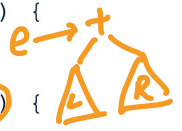
DT DT

```
public class Constant implements Expression {
  ...
  public void accept(Visitor v) {
    v.visitConstant(this);
  }
}
```

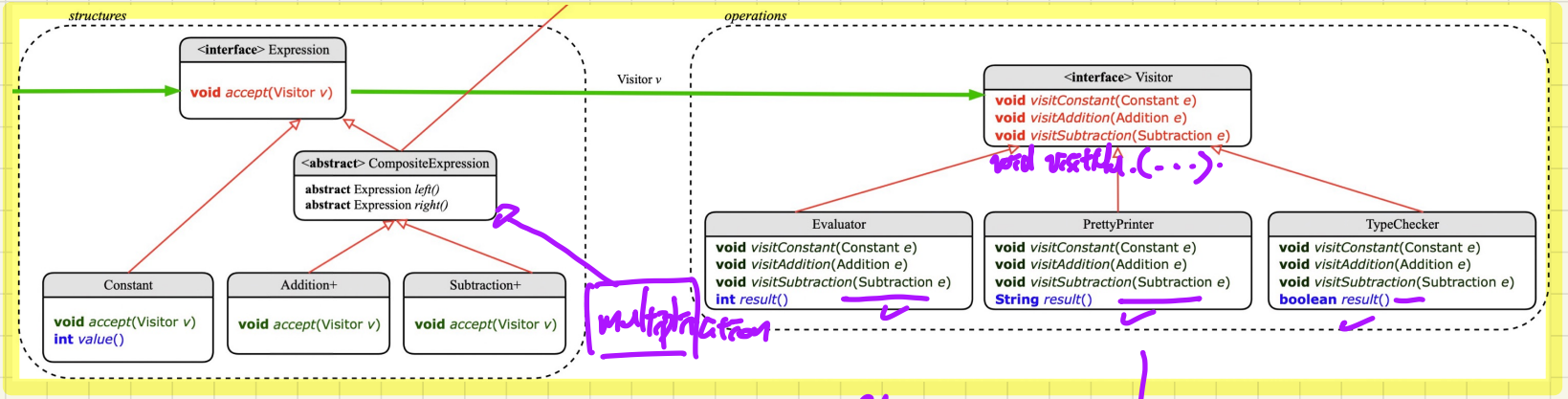
```
public class Addition extends CompositeExpression {
  ...
  public void accept(Visitor v) {
    v.visitAddition(this);
  }
}
```

```
public interface Visitor {
  public void visitConstant(Constant e);
  public void visitAddition(Addition e);
  public void visitSubtraction(Subtraction e);
}
```

```
public class Evaluator implements Visitor {
  private int result;
  ...
  public void visitConstant(Constant e) {
    this.result = e.value();
  }
  public void visitAddition(Addition e) {
    Evaluator evalL = new Evaluator();
    Evaluator evalR = new Evaluator();
    e.getLeft().accept(evalL);
    e.getRight().accept(evalR);
    this.result = evalL.result() + evalR.result();
  }
}
```



Visitor Pattern: Open-Closed and Single-Choice Principles



structure: open
operation: closed

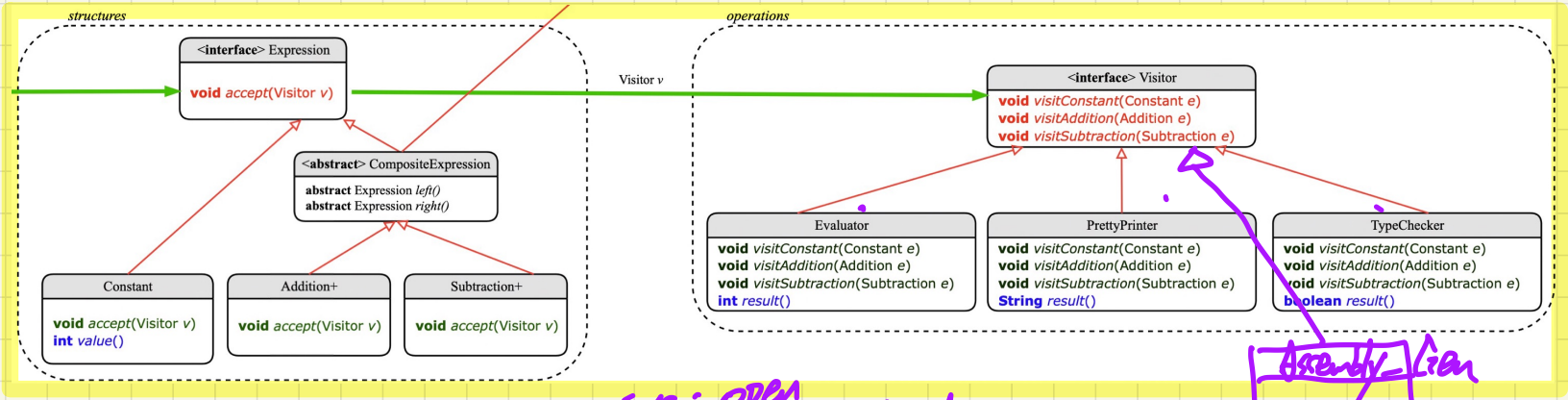
violates SCP.

What if a **new language construct** is added?

↳ unsuitable for visitor pattern

If the **visitor pattern** is adopted, what should be **closed**?

Visitor Pattern: Open-Closed and Single-Choice Principles



operations: open
structures: closed

What if a new language operation is added?

To this single place, implement all visit methods
↳ SCP satisfied

If the visitor pattern is adopted, what should be open?