



Visitor solutions

Here are the possible solutions of the implementation we asked for the Visitor Chapter ??.

1.1 Evaluator

```
EAddition >> accept: aVisitor
    ^ aVisitor visitAddition: self

EEvaluatorVisitor >> visitAddition: anEAddition
    | evaluationOfLeft evaluationOfRight |
    evaluationOfLeft := anEAddition left accept: self.
    evaluationOfRight := anEAddition right accept: self.
    ^ evaluationOfLeft + evaluationOfRight

ENegation >> accept: aVisitor
    ^ aVisitor visitNegation: self

EEvaluatorVisitor >> visitNegation: anENegation
    | aNumber |
    aNumber := anENegation expression accept: self.
    ^ aNumber negated

EMultiplication >> accept: aVisitor
    ^ aVisitor visitMultiplication: self
```

```

[EEvaluatorVisitor >> visitMultiplication: anEMultiplication

    | evaluationOfLeft evaluationOfRight |
    evaluationOfLeft := anEMultiplication left accept: self.
    evaluationOfRight := anEMultiplication right accept: self.
    ^ evaluationOfLeft * evaluationOfRight

[EDivision >> accept: aVisitor

    ^ aVisitor visitDivision: self

[EEvaluatorVisitor >> visitDivision: aDivision

    | denom numerator |
    denom := aDivision denominator accept: self.
    denom isZero
        ifTrue: [ EZeroDenominator signal ].
    numerator := aDivision numerator accept: self.
    ^ numerator / denom

[EVariable >> accept: aVisitor

    ^ aVisitor visitVariable: self

[EEvaluatorVisitor >> visitVariable: aVariable

    ^ bindings at: aVariable id

```

1.2 Printing visitor

```

[EPrinterVisitor >> visitAddition: anAddition

    | left right |
    left := anAddition left accept: self.
    right := anAddition right accept: self.
    ^ '(', left , ' + ', right, ')'

[EPrinterVisitor >> visitDivision: aDivision

    | left right |
    left := aDivision left accept: self.
    right := aDivision right accept: self.
    ^ '(', left , ' / ', right, ')'

[EPrinterVisitor >> visitNegation: aNegation

    | subExpression |
    subExpression := aNegation expression accept: self.
    ^ subExpression , ' negated'

```

1.2 Printing visitor

```
[EPrinterVisitor >> visitVariable: aVariable  
  ^ aVariable id asString
```

