CPSC 353 : More AST

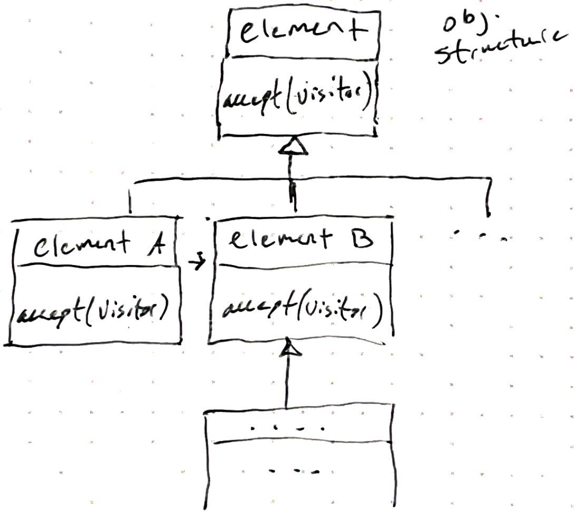
What do you do w/ AST ?

- Pretty print
- Type checking

We will impliment Visitor design pattern to seperate navigation code from the object/structure.

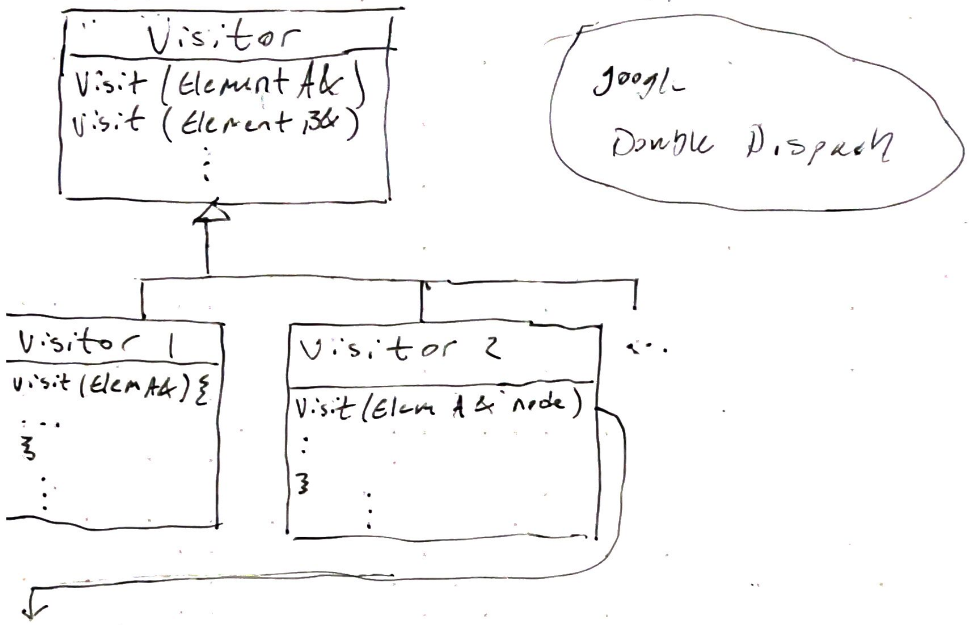
- Navigation goes into class called visitor

Visitor pattern



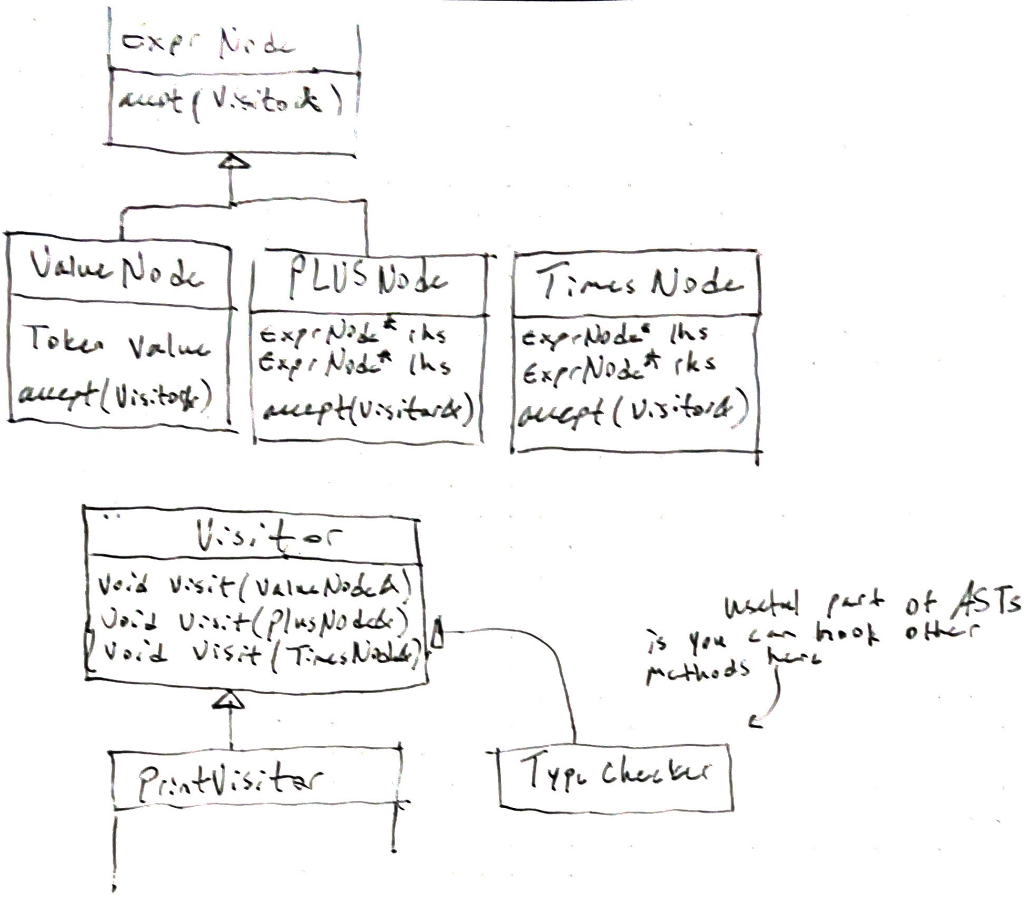
- Must define method accept(visitor v) & override method for each child method
  - v. visit(&this);

- Define visitor function for every concrete element class



need to do something w/ elem A node

```
{
    cout << element A stuff << '\n';
    for (ElementB e : elemBs
    {
        e. accept (*this);
        ...
     }
}
```



```
void visit(ValueNode& node)
{
  cout << node.value.lexeme) << "\n";
}

void visit(PlusNode& node)
{
    node.lhs -> accept(*this);
    cout << " + ";
    node.rhs -> accept(*this);

}
```