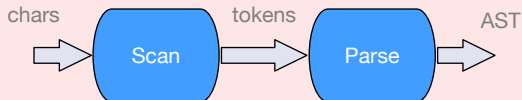


Compiler Construction

~ Pretty Printer ~

Problems statement

How to ensure that the beginning of the compiler is correct?



How to check that the translation from parse tree to ast is correct?

⇒ How to process the AST?

The pretty-printer

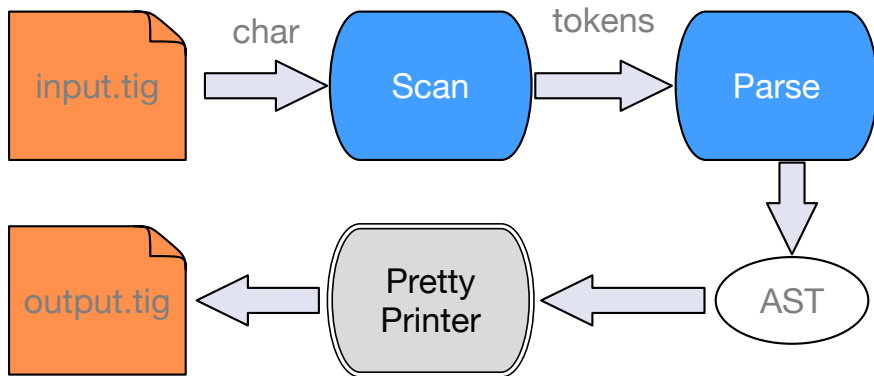
A pretty-printer converts source-code into source-code that respects a coding-style.

A coding style fixes:

- indentation
- spaces before or after parenthesis
- remove empty lines, multiple consecutive spaces
- block indentation
- ...

⇒ The pretty-printer will walk the entire AST, and produce an **equivalent but formatted output**

How does this help us?



If the input file respects the coding style, the output file must be identical!

Pretty-printer for our C++ AST (1/3)

```
class Exp
{
public:
    virtual ostream& print(ostream& o) const = 0;
};
```

Use dynamic dispatch!

Pretty-printer for our C++ AST (2/3)

```
class Num : public Exp
{
public:
    Num(int val)
        : Exp(), val_(val)
    {}

    ostream& print(ostream& o) const override
    {
        return o << val_;
    }

private:
    int val_;
};
```

Pretty-printer for our C++ AST (3/3)

```
class Bin : public Exp {
public:
    /* Ctors, Dtors & Getters */

    ostream& print(ostream& o) const override
    {
        o << '('; lhs_->print(o);
        o << oper_;
        rhs_->print(o);
        return o << ')';
    }
private:
    char oper_;
    Exp* lhs_; Exp* rhs_;
};
```

Testing our pretty-printer

```
int main()
{
    Bin* bin = new Bin(
        '+',
        new Num(42),
        new Num(51)
    );

    Exp* exp = bin;
    bin.print(cout)
    exp.print(cout)
    delete bin;
}
```

(42+51)

(42+51)

Summary

