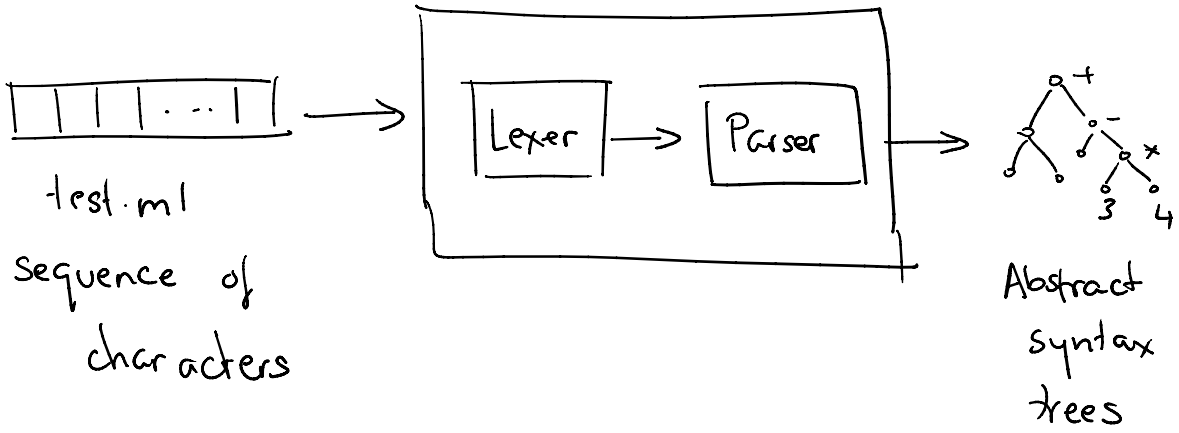
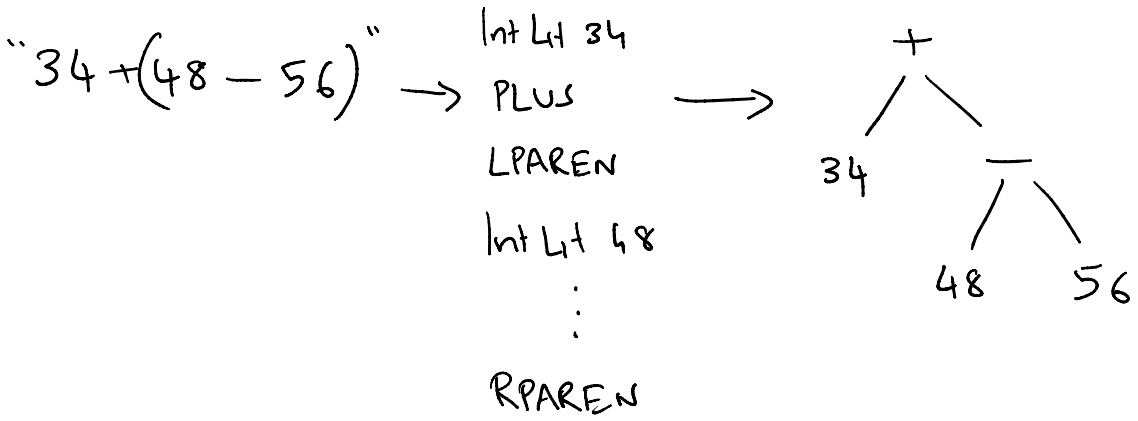
Lecture 13

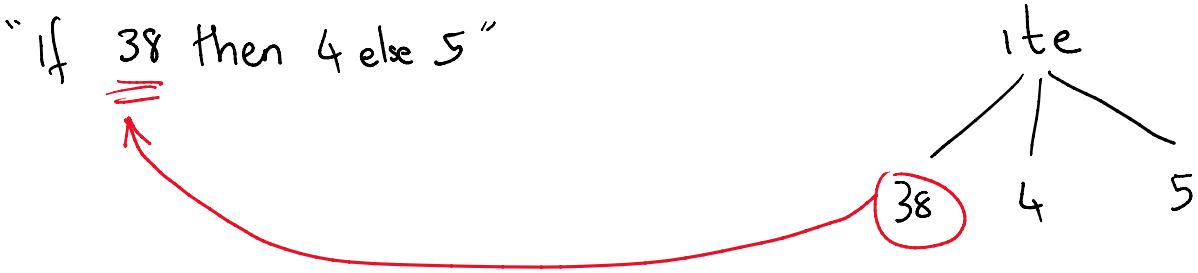
Tuesday, October 5, 2021 2:00 PM



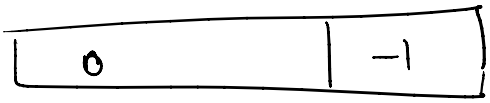
Sequence of characters

Abstract syntax trees





$\Rightarrow (1 + \text{mantissa}) \cdot 2^{\text{exponent}}$



$0.5 = 2^{-1}$

Describing the syntax of arithmetic expressions

(Attempt 1)

```
expr:
  | INT
  | expr; PLUS; expr
  | expr; MINUS; expr
  | LPAREN; expr; RPAREN
```

Non-terminal symbols

```
Expr ::=
  | Int
  | Expr '+' Expr
  | Expr '-' Expr
  | '(' Expr ')'
```

Production rule

Recipe to produce all valid expressions that we can provide to the calculator

```
type expr =
  | Int of int
  | Plus of expr * expr
  | Minus of expr * expr
```

Expression as it lives in the tool

lives in the text file / string

| Minus of expr * expr

Expression as it lives in our interpreter (which is itself written in Ocaml.)

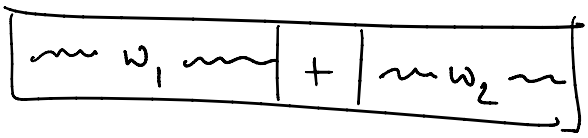
```
Expr ::= INT
       | Expr '+' Expr
       | Expr '-' Expr
       | '(' Expr ')'
```

① Sequence of characters

② Parser tells us that it is well formed.

③ In addition it is

well-formed because:



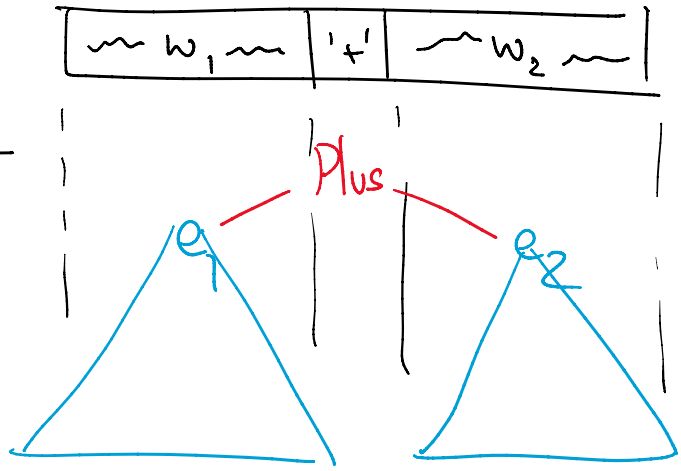
a. There is a '+' somewhere in the middle

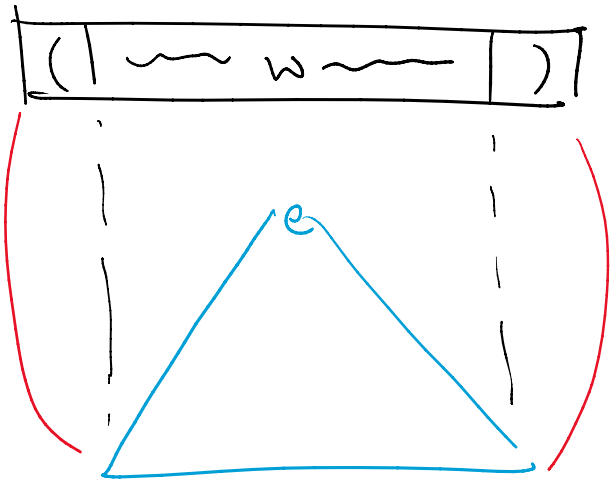
b. Both substrings w_1 & w_2 to the left & right are well-formed.

c. In other words, there must be ASTs e_1 & e_2

String world

Ocaml world





$3+4-5$

$(3+4)-5$

or

$3+(4-5)$

The AST associated with the full string

"(w)"

is the same as the AST associated with the substring

"w"