

CSC215

Math and Computer Science



2 Problems to Consider

- Translation
 - Infix \leftrightarrow prefix
 - Infix \leftrightarrow postfix
- Evaluation
 - Check for valid expression
 - Evaluate the expression if valid
- We will look at these 2 problems using recursion and stacks

Validation of Prefix

$\langle \text{prefix} \rangle = \langle \text{identifier} \rangle \mid \langle \text{operator} \rangle \langle \text{prefix} \rangle \langle \text{prefix} \rangle$

$\langle \text{identifier} \rangle = a - z \mid 0 - 9$

$\langle \text{operator} \rangle = + - * /$

This is how grammars are specified.

BNF (Backus Normal Form or Backus–Naur Form)

a-z, 0-9 are literals

$\langle \rangle$ are expressions that need more evaluation

Simple expression - 1

+35

Is this a valid prefix

+35

Is the 1st character an identifier, no



Is it an operator? Yes, then if we can find 2 identifiers it is a valid prefix by the previous rules.

Move to the next character

Simple expression -1

+35
↑

Is it an Identifier? Yes

3 is a valid prefix

Move to the next character

+35
↑

Is it an identifier? Yes

5 is a valid prefix

+35
↑

move to the next character, end of expr.

Valid expression, found 2 prefix expr.

for operator

Medium Expression - 2

+*abc
↑

Look at first character, is it an identifier? No
Is it an operator, yes. Need to find 2 prefix
Move to next character

+*abc
↑

Is it an identifier? No
Is it an operator, yes. Need to find 2 prefix
Move to next character.

Medium Expression - 2

+*abc
↑

Is it an identifier? Yes

Found the 1st prefix for the *

Move to next character

+*abc
↑

Is it an identifier? Yes

Found the 2nd prefix for the *

*ab is a valid prefix which satisfies being an
prefix for the 1st operand of the +

Move to the next character

Medium Expression - 2

Start looking for 2nd prefix where the 1st prefix left off at in expression

+*abc



Is it an identifier? Yes

Found the 2nd prefix for the +
making the +*abc a valid prefix

Move to next character

+*abc



Used all character it is valid

Harder Expression - 3

*+-ab*cde
↑

Look at first character

Is it an identifier? No

Is it an operator? Yes, need to find 2 prefix
to make it a valid prefix.

Move to next character

Harder Expression - 3

*+-ab*cde
↑

Is it an identifier? No

Is it an operator? Yes, need to find 2 prefix
to make it a valid prefix.

Move to next character

Harder Expression - 3

*+-ab*cde



Is it an identifier? No

Is it an operator? Yes, need to find 2 prefix
to make it a valid prefix.

Move to next character

Harder Expression - 3

*+-ab*cde



Is it an identifier? Yes

Found 1st valid prefix for the - operator.

Look for 2nd prefix where the 1st prefix ended.

Harder Expression - 3

*+-ab*cde
↑

Is it an identifier? Yes

Found 2nd valid prefix for the - operator.

-ab is a valid prefix which makes it the 1st prefix for the + operator.

Start looking 2nd prefix where the 1st left off

Harder Expression - 3

*+-ab*cde



Is it an identifier? No

Is it an operator? Yes , need to find 2 prefix
to make it a valid prefix.

Move to next character

Harder Expression - 3

*+-ab*cde



Is it an identifier? Yes

Found 1st valid prefix for the * operator, need to find the 2nd prefix starting where the first prefix left off

Harder Expression - 3

*+-ab*cde
↑

Is it an identifier? Yes

Found 2nd prefix for the * operator. That makes the *cd a valid prefix and the 2nd prefix for the + (-ab was first identifier).

This make +-ab*cde a valid prefix and the 1st prefix for the *. Start looking for the 2nd prefix where the first left off.

Harder Expression - 3

*+-ab*cde
↑

Is it an identifier? Yes

Found 2nd prefix for the * operator.

+ - ab * cd was the 1st prefix

e was the 2nd prefix

that make *+-ab*cde a valid prefix

Move to next character

Harder Expression - 3

*+-ab*cde



All characters have been used
The entire expression is valid

Invalid Expression - 4

-+ab



Look at first character

Is it an identifier? No

Is it an operator? Yes, find 2 valid prefix expressions.

Move to next character

Invalid Expression - 4

-+ab



Is it an identifier? No

Is it an operator? Yes, find 2 valid prefix expressions.

Move to next character

Invalid Expression - 4

-+ab



Is it an identifier? Yes, this is the 1st valid prefix for the + operator.

Start looking for the 2nd prefix where the first left off.

Invalid Expression - 4

-+ab
↑

Is it an identifier? Yes, this is the 2nd valid prefix for the + operator.

This makes +ab a valid prefix for the – operator. Start looking for the 2nd prefix where the 1st prefix left off.

Invalid Expression - 4

-+ab



Short a prefix, not a valid expression

Invalid Expression - 5

+abc



Look at first character

Is it an identifier? No

is it an operator? Yes, must find 2 valid prefix expressions to make it valid.

Move to next character

Invalid Expression - 5

+abc



Is it an identifier? Yes, this is the 1st valid prefix for the + operator. Start looking for the 2nd where the first expression ended.

Invalid Expression - 5

+abc



Is it an identifier? Yes, this is the 2nd valid prefix for the + operator.

This makes the +ab a valid prefix expression.

Move to the next character

Invalid Expression - 5

+abc
↑

There are excess characters, this makes the entire expression invalid.

Invalid Expression - 6

<ab



Look at the first character

Is it an identifier? No

Is it an operator? No

Invalid character in expression.

Algorithm for validating prefix

- If $s[\text{first}]$ is an operand (identifier) return first
- Else if $s[\text{first}]$ is an operator
 - Check that $s[\text{first}+1 \dots \text{end1}]$ is a valid prefix expression
 - If it is a valid prefix, check that $s[\text{end1}+1 \dots \text{end2}]$ is valid, return end2
 - Else return -1
- $s[\text{first}]$ is not a valid character return -1
- Must use all character in the expression

Validation of Postfix

$\langle \text{postfix} \rangle = \langle \text{identifier} \rangle \mid \langle \text{postfix} \rangle \langle \text{postfix} \rangle \langle \text{operator} \rangle$

$\langle \text{identifier} \rangle = a - z \mid 0 - 9$

$\langle \text{operator} \rangle = + - * /$

- Start at the end and work towards the beginning.
- Same algorithm as validating a prefix expression.

Simple Expression - 1

ab+
↑

Look at last character

Is it an identifier? No

Is it an operator? Yes, find 2 valid postfix expressions to make it valid.

Back up one character

Simple Expression - 1

ab+
↑

Is it an identifier? Yes, this is the 1st postfix expression for the + operator.

Start looking for the 2nd postfix where the first one left off.

Simple Expression -1

ab+



Is it an identifier? Yes, this is the 2nd postfix expression for the + operator.

This makes ab+ a valid postfix

Back up one character

Steps off the front, all characters used
and ab+ is a valid postfix

Medium Expression - 2

abc*+
↑

Look at last character

Is it an identifier? No

Is it an operator? Yes, must find 2 valid postfix expressions for it to be valid.

Back up one character.

Medium Expression - 2

abc*+



Is it an identifier? No

Is it an operator? Yes, must find 2 valid postfix expressions for it to be valid.

Back up one character.

Medium Expression - 2

abc*+



Is it an identifier? Yes, this is the 1st valid postfix expression for the * operator
Start looking for the 2nd postfix where the first left off.

Medium Expression - 2

abc*+



Is it an identifier? Yes, this is the 2nd valid postfix expression for the * operator.

This makes bc* the 1st valid postfix expression for the + operator.

Start looking for the 2nd valid postfix where the 1st expression left off.

Medium Expression - 2

abc*+



Is it an identifier? Yes, this is the 2nd valid postfix expression for the + operator.

bc* 1st valid postfix

a 2nd valid postfix

back up one character and we step off the front of the expression. All characters have been used.

Harder Expression - 3

ab-cd*+e/
↑

Look at last character

Is it an Identifier? No

Is it an operator? Yes, need to find 2 valid postfix expressions for this to be valid.

Back up one character.

Harder Expression - 3

ab-cd*+e/
↑

Is it an Identifier? Yes

Found the first valid postfix expression
for the / operator.

Start looking where the first ended.

Harder Expression - 3

ab-cd*+e/



Is it an Identifier? No

Is it an operator? Yes, must find 2 valid postfix expressions for this to be valid.

Back up one character.

Harder Expression - 3

ab-cd*+e/



Is it an Identifier? No

Is it an operator? Yes, must find 2 valid postfix expressions for this to be valid.

Back up one character.

Harder Expression - 3

ab-cd*+e/



Is it an Identifier? Yes, this is the 1st valid postfix for the * operator.

Start looking for the 2nd where the first expression left off.

Harder Expression - 3

ab-cd*+e/



Is it an Identifier? Yes, this is the 2nd valid postfix for the * operator.

cd* is the 1st valid postfix expression for the + operator.

Start looking for the 2nd valid postfix where the first left off.

Harder Expression - 3

ab-cd*+e/



Is it an Identifier? No

Is it an operator? Yes, need to find 2 valid postfix for this to be valid.

Back up one character.

Harder Expression - 3

ab-cd*+e/



Is it an Identifier? Yes, this is the 1st valid postfix expression for the – operator
Start looking for the 2nd postfix where the first left off.

Harder Expression - 3

ab-cd*+e/



Is it an Identifier? Yes, this is the 2nd valid postfix expression for the – operator
ab- is the 2nd valid postfix for the + operator.

ab-cd*+ is the second valid postfix for the / operator making ab-cd*+e/ a valid postfix expression.

Harder Expression - 3

ab-cd*+e/



Stepped off the front.

All characters used and produced a
valid postfix expression

Invalid Expression - 4

abc+
↑

Look at last character

is it an identifier? No

Is it an operator? Yes, must find 2 valid postfix expressions for it to be valid.

Back up one character.

Invalid Expression - 4

abc+
↑

is it an identifier? Yes, this is the 1st valid postfix expression for the + operator.
Start looking for the 2nd postfix where the 1st left off.

Invalid Expression - 4

abc+
↑

is it an identifier? Yes, this is the 2nd valid postfix expression for the + operator.

This makes bc+ a valid postfix.

Back up one character

Have an extra character that is not needed, the expression is invalid.

Invalid Expression - 5

b+
↑

Look at last character

Is it an identifier? No

Is it an operator? Yes, must find 2 valid postfix expressions to make this valid.

Back up one character.

Invalid Expression - 5

b+



Look at last character


Is it an identifier? Yes, this is the 1st valid postfix expression for the + operator.

Start looking for the 2nd valid postfix expression where the 1st left off.

Back up one character.

Invalid Expression - 5

b+



Stepped off the front

This expression is invalid because there is not a 2nd postfix expression for the + operator.

Invalid Expression - 6

ab>

Look at the last character

Is it an identifier? No

Is it an operator? No

Invalid character in expression.

Algorithm for Validating postfix

- If $s[\text{last}]$ is an operand (identifier) return last
- Else if $s[\text{last}]$ is an operator
 - Check that $s[\text{last}-1 \dots \text{first1}]$ is a valid postfix expression
 - If it is a valid postfix, check that $s[\text{first1}-1 \dots \text{first2}]$ is valid, return first2
 - Else return -1
- $s[\text{last}]$ is not a valid character return -1
- Must use all character in the expression