
Gianni Morandi
Dashboard
Compiler
Algorithms
CC Assistant
Assignments
Exercises

Dashboard
Compiler
Algorithms
CC Assistant
Assignments
Exercises

Lexeme	Regular expression
WHITESPACE	**[\t\r\n\u00C]+**
IntegerLiteral	**(-|+)?[0-9]+**
BooleanLiteral	**true|false**

+

PREVIOUS
UPLOAD TOKENS
NEXT

(- | +) ? [0 - 9] +

Regular expression parse tree

ADD NODE

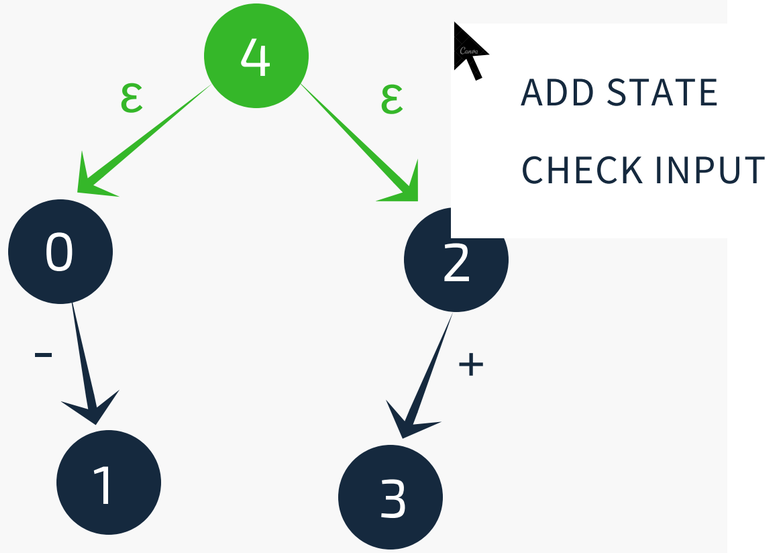
CHECK INPUT

NFA

(- | +) ? [0 - 9] +

Regular expression parse tree

NFA



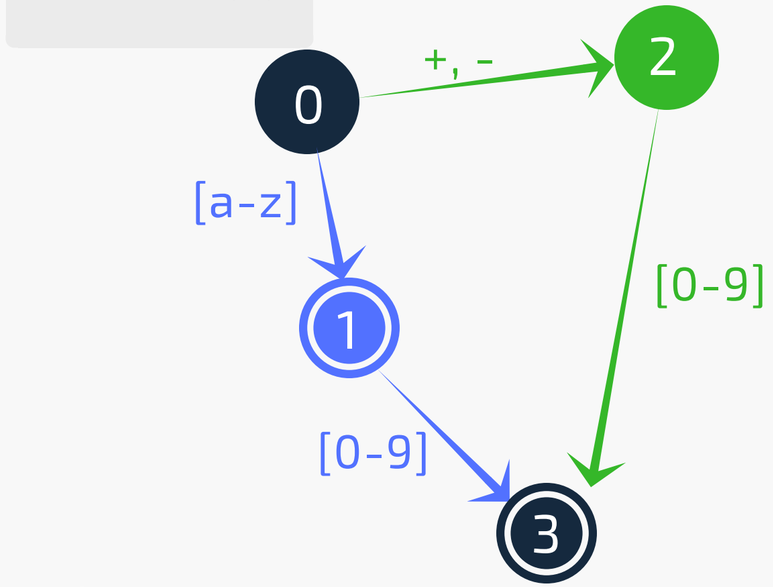
Grammar

```
program
    mainClass, { classDeclaration }, EOF

mainClass
    CLASS, Identifier, LBRACE, PUBLIC,
    STATIC,
    VOID, "main", LPAREN, RPAREN,
     LBRACE, statement, RBRACE, RBRACE
```

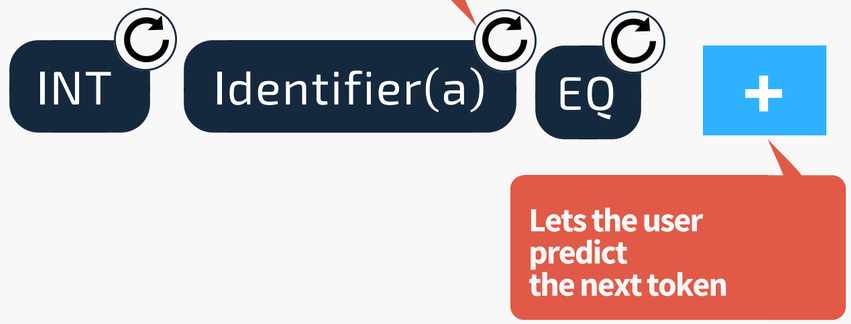
Text to parse

```
/**
 * Create an input stream from a socket.
 */
public In(java.net.Socket socket) {
    try {
        InputStream is = socket.getInputStream();
        scanner = new Scanner(new BufferedInputStream(is), CHARSET_NAME);
        scanner.useLocale(LOCALE);
    }
    catch (IOException ioe) {
        System.err.println("Could not open " + socket);
    }
}

/**
 * Create an input stream from a URL.
 */
public In(URL url) {
    try {
        URLConnection site = url.openConnection();
        InputStream is     = site.getInputStream();
        scanner            = new Scanner(new BufferedInputStream(is), CHARSET_NAME);
        scanner.useLocale(LOCALE);
    }
    catch (IOException ioe) {
        System.err.println("Could not open " + url);
    }
}
```

DFA



Generated tokens



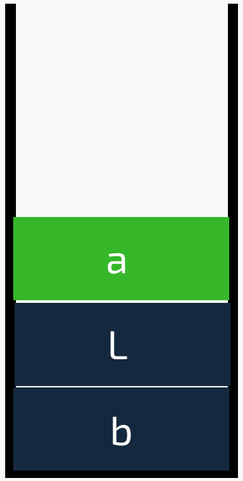
Change simulation settings. E.g. speed, whether to show the debugging page, etc...

BACK | PREVIOUS | NEXT | SKIP

CURRENT ACTION: **MATCH a**

Tokens

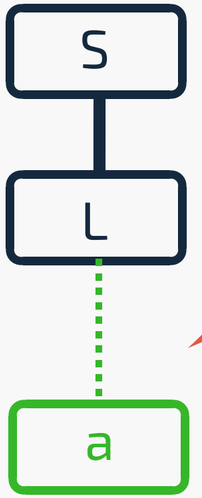
Stack



Grammar

S -> Lb
L -> aL
L -> ε

Parse tree



This is the addition that will be visualized after doing a match (i.e. pressing next)

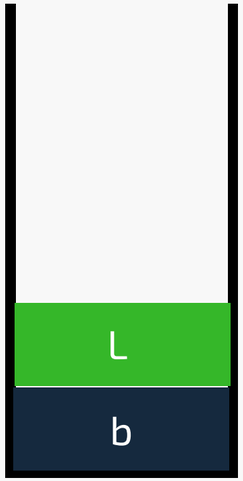
Compact visualization

BACK | PREVIOUS | NEXT | SKIP

CURRENT ACTION: **PREDICT L -> aL**

Tokens

Stack



Grammar

S -> Lb
L -> aL
L -> ε

Parse tree

COMPUTING FIRST(aL)

After computing the FIRST set of 'aL', we check if the current character (a) is in this set and report to the user.

FIRST(aL) = {a}

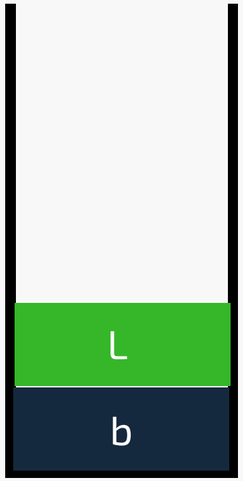
Compact visualization

BACK
PREVIOUS
NEXT
SKIP

CURRENT ACTION: **PREDICT L -> aL**

Tokens

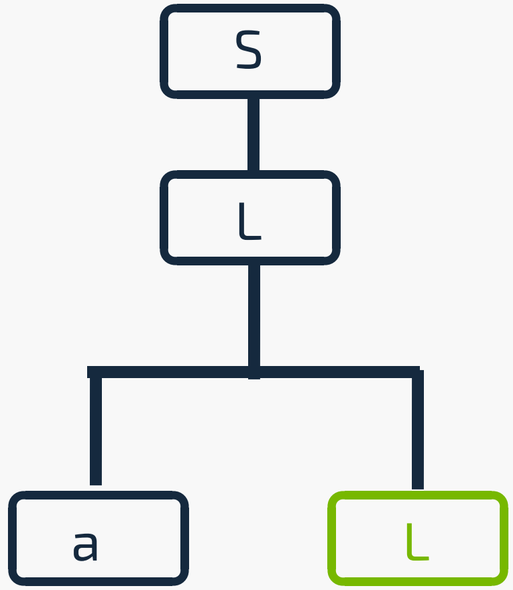
Stack



Grammar

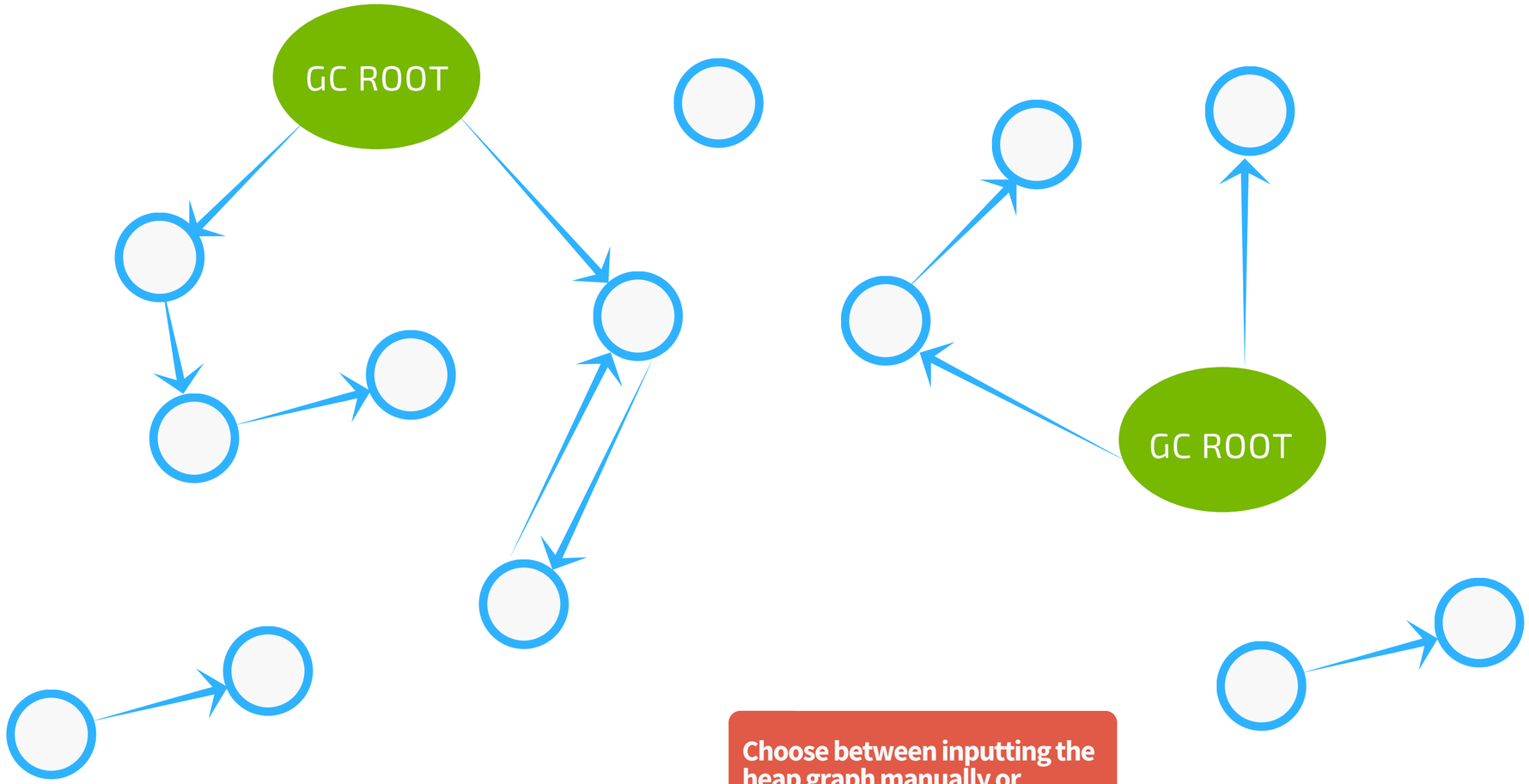
S -> Lb
L -> aL
L -> ε

Parse tree



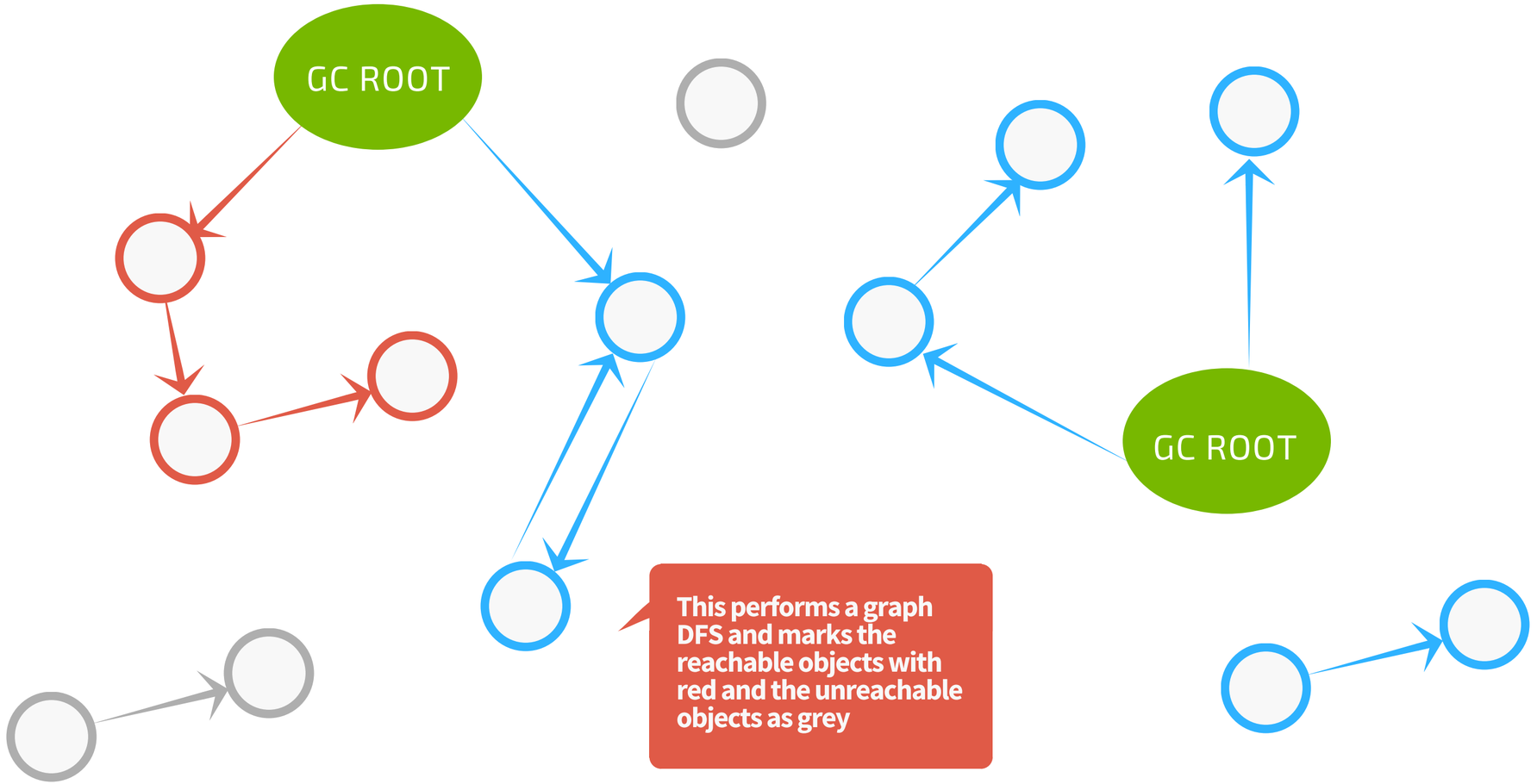
Compact visualization

BACK
PREVIOUS
NEXT
SKIP



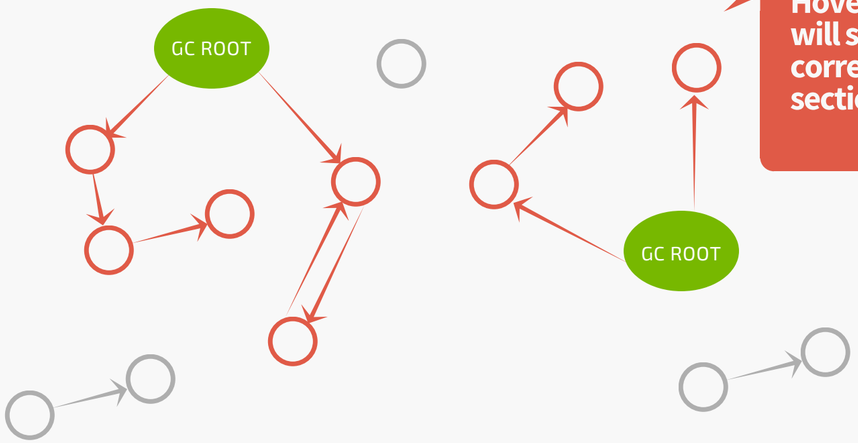
Choose between inputting the heap graph manually or generating it from Java code

PREVIOUS | HEAP GRAPH | NEXT

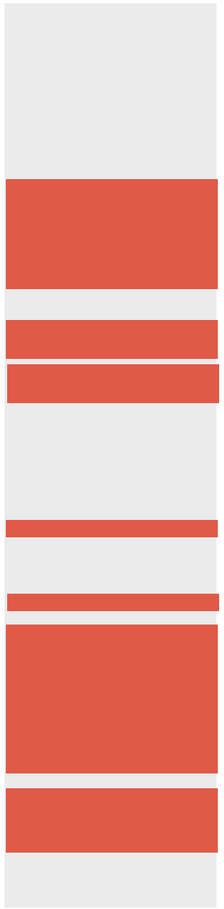


PREVIOUS
NEXT

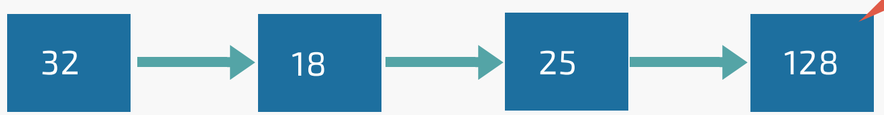
Heap graph



Hovering over the nodes will show the corresponding section in the heap



Free list



The numbers indicate the size of the free chunks

Gianni Morandi

- Dashboard
- Compiler
- Algorithms
- CC Assistant
- Assignments
- Exercises

CONTINUE DEVELOPMENT

Syntax Analysis

Currenlty on: Building AST.

Resume (88%)

SUGGESTED

Garbage Collection

Start

ASSIGNMENTS

NFA to DFA (Resume)

Grammar Left Factoring (Start)

Regex to NFA (Start)

CUSTOM TEST CASES

Regex to NFA

Edit Test Delete

Regex to NFA

Author: John Smith

Edit Test

This is a test case submitted by the teacher

Gianni Morandi

Learn

Learn

Transition.java | State.java | NFA.java

```
1
2
```