



ΜΕΤΑΓΛΩΤΙΣΤΕΣ

2^η Εργασία : Κατασκευή Αφηρημένου Δέντρου και Σημασιολογική Ανάλυση της γλώσσας MiniPython

| Ονοματεπώνυμο | Αριθμ. Μητρώου | E-mail επικοινωνίας |
|-----------------------|----------------|---------------------|
| Γεώργιος Μοσχόβης | 3150113 | p3150113@aueb.gr |
| Γιαννακόπουλος Θάνος | 3150234 | p3150234@aueb.gr |
| Κουλούρης Κωσταντίνος | 3120081 | p3120081@aueb.gr |
| Τρογκάνη Παναγιώτα | 3120187 | p3120187@aueb.gr |

Στην παρούσα φάση της εργασίας, κληθήκαμε, πρώτον, να δημιουργήσουμε το Αφηρημένο Συντακτικό Δέντρο της γραμματικής με σκοπό τη μείωση του μεγέθους του συντακτικού δέντρου ενός προγράμματος σε minipython και δεύτερον, τη σημασιολογική ανάλυση του ίδιου προγράμματος με βάση το αφηρημένο συντακτικό δέντρο που δημιουργήσαμε παραπάνω.

Για την κατασκευή του αφηρημένου συντακτικού δέντρου, εκμεταλλευτήκαμε τις ιδιότητες του εργαλείου SableCC. Η γραμματική μας καταφέρνει και τηρεί, όπως και στο πρώτο μέρος, τις προτεραιότητες σε όλες τις πράξεις. Παραθέτουμε ορισμένα παραδείγματα αφηρημένων συντακτικών δέντρων που δημιουργήθηκαν από την εφαρμογή μας.

Μετά χρησιμοποιήσαμε Visitors για να διασχίσουμε το ΑΣΔ. Συνοπτικά, γράψαμε ένα Data Access Object (DAO) για να διατηρεί τις δηλώσεις συναρτήσεων και δυο τάξεις επισκέπτη που κληρονομούν (extends) την DepthFirstAdapter που βρίσκεται στον κατάλογο MiniPython/analysis όπου ακυρώσαμε (override) όσες μεθόδους της DepthFirstAdapter δε χρειαστήκαμε στην κατεύθυνση της συμπλήρωσης του πίνακα συμβόλων και της ανίχνευσης λαθών κατά την διαδικασία αυτή.

Κατασκευή αφηρημένου συντακτικού δέντρου

Παράδειγμα 1^ο

```
example1.py x
1 def x(a, k):
2     return a + k
3
4 k = "0"
5
6 print x(0, x(0,1+1))
```

```

>- AProgram
|
| - AFunctionGoal
|   ^- AFunction
|     | - x
|     |   | - ASimpleArgumentArgument
|     |   |   ^- a
|     |   | - ASimpleArgumentArgument
|     |   |   ^- k
|     |   ^- AReturnStatement
|     |     ^- AAdditionExpression
|     |       | - AIdentifierExpression
|     |       |   ^- a
|     |       | - AIdentifierExpression
|     |       |   ^- k
|
| - AStatementGoal
|   ^- AAssignStatement
|     | - k
|     ^- AValueExpression
|       ^- AStringValue
|         ^- "0"
|
| - AStatementGoal
|   ^- APrintStatement
|     ^- AFuncallExpression
|       ^- AFuncall
|         | - x
|         |   | - AValueExpression
|         |   |   ^- ANumberValue
|         |   |   ^- 0
|         |   ^- AFuncallExpression
|         |     ^- AFuncall
|         |       | - x
|         |       |   | - AValueExpression
|         |       |   |   ^- ANumberValue
|         |       |   |   ^- 0
|         |       |   ^- AAdditionExpression
|         |       |     | - AValueExpression
|         |       |     |   ^- ANumberValue
|         |       |     |   ^- 1
|         |       |     ^- AValueExpression
|         |       |       ^- ANumberValue
|         |       |       ^- 1

```

Πρόγραμμα σε Minipython

Αφηρημένο Συντακτικό Δέντρο

Παράδειγμα 2^ο – Εφαρμογή προτεραιότητας πράξεων

```
1  
2 x = (1 + 1) + 9 * 8 + 9
```

```
>- AProgram
  ^- AStatementGoal
    ^- AAssignStatement
      |- x
      ^- AAdditionExpression
        |- AAdditionExpression
          | | |- AAdditionExpression
          | | | | |- AValueExpression
          | | | | ^- ANumberValue
          | | | | ^- 1
          | | | ^- AValueExpression
          | | | ^- ANumberValue
          | | | ^- 1
          | ^- AMultiplicationExpression
          | | |- AValueExpression
          | | | ^- ANumberValue
          | | | ^- 9
          | ^- AValueExpression
          | ^- ANumberValue
          | ^- 8
        ^- AValueExpression
          ^- ANumberValue
          ^- 9
```

Πρόγραμμα σε Minipython

Αφηρημένο Συντακτικό Δέντρο

Επίδειξη δυνατοτήτων μεταγλώττισης MiniPython

C:\Users\George\Desktop\Compilers\paradoteo\faultyCode1.py - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run



faultyCode1.py

```
1 def add(x,y):
2     return x + y
3 print k
```

C:\Windows\System32\cmd.exe

```
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest faultyCode1.py
Variable k in line: 2 is not defined and can't be used here!
```

*C:\Users\George\Desktop\Compilers\paradoteo\faultyCode2.py - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run



faultyCode2.py

```
1 def add(x,y):
2     return x + y
3 print k
4 k=0
```

C:\Windows\System32\cmd.exe

```
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest faultyCode2.py
Variable k in line: 2 is not defined and can't be used here!
```

*C:\Users\George\Desktop\Compilers\paradoteo\faultyCode3.py - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run



faultyCode3.py

```
1 def add(x,y):
2     return x + y
3 k="hello world"
4 print add(2,k)
```

C:\Windows\System32\cmd.exe

```
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest faultyCode3.py
Error in line 5: Addition is allowed only for numbers or strings!
```

Όλα τα παραπάνω αρχεία παρατίθενται στον υποκατάλογο examples του παραδοτέου.

```
*C:\Users\George\Desktop\Compilers\paradoteo\faultyCode4.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
faultyCode4.py
1 def add(x,y):
2     return "hello world"
3     print add(2,1)+2
```

```
C:\Windows\System32\cmd.exe
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest faultyCode4.py
Error in line 4: Addition is allowed only for numbers or strings!
C:\Users\George\Desktop\Compilers\paradoteo>
```

```
*C:\Users\George\Desktop\Compilers\paradoteo\faultyCode5.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
faultyCode5.py
1 def add(x,y):
2     return "hello world"
3     print add(2,1)*2
4
```

```
C:\Windows\System32\cmd.exe
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest faultyCode5.py
Error in line: 4 Substraction/Multiplication/Division is only permitted for decimals or integers
C:\Users\George\Desktop\Compilers\paradoteo>
```

```
*C:\Users\George\Desktop\Compilers\paradoteo\faultyCode6.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
faultyCode6.py
1 def add(x,y):
2     return "hey!"
3
4 def add(x,y,z=1):
5     return "hey again"
```

```
C:\Windows\System32\cmd.exe
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest faultyCode6.py
Method add in line: 5 already defined
C:\Users\George\Desktop\Compilers\paradoteo>
```

```
*C:\Users\George\Desktop\Compilers\paradoteo\faultyCode7.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
faultyCode7.py
1 def add(x,y,z):
2     return "hey!"
3
4 def add(x,y,z=1):
5     return "hey again!"
```

```
C:\Windows\System32\cmd.exe
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest faultyCode7.py
Method add in line: 5 already defined
C:\Users\George\Desktop\Compilers\paradoteo>
```

```
*C:\Users\George\Desktop\Compilers\paradoteo\rightCode1.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
rightCode1.py
1 def add(x,y=2):
2     return x + y
3     print add(1)
```

Εγκυρά αρχεία MiniPython

```
*C:\Users\George\Desktop\Compilers\paradoteo\rightCode2.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
rightCode2.py
1 def g(a, k = 1):
2     return a - k
3
4 def f(x):
5     return x
6
7     x = 5
8     print f(g(x) + 3.0)
9     x = 2.0 - f(2)
10    print g(x)
11
```

```
C:\Windows\System32\cmd.exe
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest rightCode1.py
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest rightCode2.py
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest rightCode3.py
C:\Users\George\Desktop\Compilers\paradoteo>java ParserTest rightCode4.py
C:\Users\George\Desktop\Compilers\paradoteo>
```

```
*C:\Users\George\Desktop\Compilers\paradoteo\rightCode3.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
rightCode3.py
1 def g(a):
2     return "hello " + a
3
4 def f(x):
5     return x
6
7     x = 5
8     print f(g("George") + " from Compiler")
9     print g(x)
```

```
*C:\Users\George\Desktop\Compilers\paradoteo\rightCode4.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run
rightCode4.py
1 def g(a = "world"):
2     return "hello " + a
3
4 def h(a, k = 1):
5     return 5 + a
6
7 def f(x):
8     return x
9
10    x = "Greeting"
11    x = x + f(g("George") + " from compiler")
12    print f(x)
```

Enjoy our compiler!