

Θέμα 1 [2 μονάδες]

Δίνεται η ακόλουθη γραμματική:

<pre><assign> -> <id> = <expr> <id> -> A B C <expr> -> <expr> + <term> <term> <term> -> <term> * <factor> <factor> <factor> -> (<expr>) <id></pre>

Γράψτε μια αριστερότερη παραγωγή (leftmost derivation) και σχεδιάστε το αντίστοιχο δένδρο συντακτικής ανάλυσης (parse tree) για την πρόταση: **A = A * (B + C)**

Θέμα 2 [A = 2 μονάδες, B = 1 μονάδα]

A. Δίνονται οι ακόλουθοι κώδικες στη γλώσσα C:

<pre>#include <stdio.h> void fun() { static int a = 0; extern int x; { int x = 0; a++; x++; printf("%d. x = %d\n", a, x); } a++; x++; printf("%d. x = %d\n", a, x); } int main() { fun(); fun(); }</pre>	<pre>int x = 5;</pre>
Περιεχόμενα του αρχείου a.c	Περιεχόμενα του αρχείου b.c

1. Τι μήνυμα σφάλματος θα εμφανίσει η ακόλουθη εντολή μεταγλώττισης και γιατί;
\$ gcc a.c
2. Ποια εντολή μεταγλώττισης πρέπει να δοθεί έτσι ώστε να μην υπάρχουν λάθη κατά τη μεταγλώττιση και σύνδεση, και τι θα εμφανιστεί κατά την εκτέλεση σε αυτή την περίπτωση;

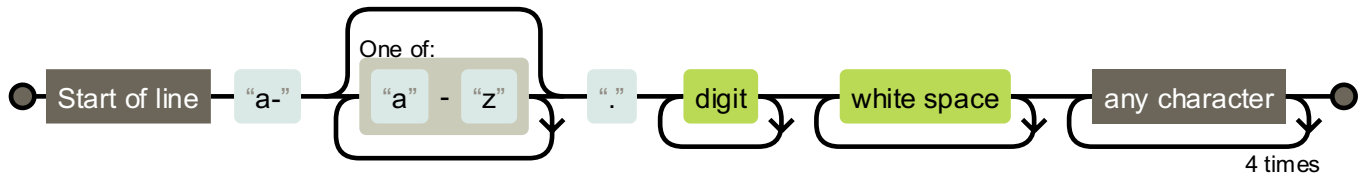
B. Τι θα εμφανίσουν κατά την εκτέλεση τα ακόλουθα τμήματα κώδικα σε Python; Αν κάποιος κώδικας εμφανίζει UnboundLocalError αναφέρατε τη μεταβλητή και το λόγο για τον οποίο προκύπτει σφάλμα.

<pre>x = 5 def fun(y): print(f"{x=}, {y=}") y += 1 print(f"{x=}, {y=}") z = 0 fun(z) print(f"{x=}, {z=}")</pre>	<pre>x = 5 def fun(y): print(f"{x=}, {y=}") y += 1 x += 1 print(f"{x=}, {y=}") z = 0 fun(z) print(f"{x=}, {z=}")</pre>	<pre>x = 5 def fun(y): global x print(f"{x=}, {y=}") y += 1 x += 1 print(f"{x=}, {y=}") z = 0 fun(z) print(f"{x=}, {z=}")</pre>
(i)	(ii)	(iii)

Θέμα 3 [A = 1 μονάδα, B = 1 μονάδα]

A.

1. Γράψτε την κανονική έκφραση που αντιστοιχεί στο ακόλουθο διάγραμμα:



2. Γράψτε 1 λεκτικό που θα εντόπιζε η συγκεκριμένη κανονική έκφραση. Δίνεται ότι:

- ^ έναρξη γραμμής
- . οποιοσδήποτε χαρακτήρας
- \d ψηφίο
- \s λευκός χαρακτήρας (π.χ. διάστημα, tab)

B. Συμπληρώστε κατάλληλες τιμές, στη θέση των ερωτηματικών ?, στο unittest που ακολουθεί έτσι ώστε να περνά όλους τους ελέγχους.

```
import unittest

def collatz(n):
    if n % 2 == 0:
        return n / 2
    else:
        return 3 * n + 1

class TestMidterm(unittest.TestCase):
    def test1(self):
        self.assertEqual(collatz(10), ?)
    def test2(self):
        self.assertEqual(collatz(7), ?)
    def test3(self):
        self.assertEqual(collatz(15), ?)

unittest.main()
```

Θέμα 4 [A = 2 μονάδες, B = 1 μονάδα]

A. Δίνεται η ακόλουθη συνάρτηση σε Haskell:

```
foo :: Double -> Double -> Double -> Double -> Double
foo x1 y1 x2 y2 = ((x1 - x2) ^ 2 + (y1 - y2) ^ 2) ** 0.5
```

1. Τι σημαίνει η πρώτη γραμμή της συνάρτησης;
2. Τι πιστεύετε ότι κάνει η συνάρτηση και τι θα επιστρέψει η κλήση της ως εξής:

```
foo 0 0 3 4
```

B. Ποια θα είναι τα αποτελέσματα των ακόλουθων ερωτημάτων στην Prolog;

1. ?- A = 1, A = A + 1.
2. ?- [_,_,_|T] = [1,2,3,4,5].
3. ?- append(L1, [5,6,7], [1,2,3,4,5,6,7]).
4. ?- A = 1, B = 2, D = A + B + 3
5. ?- X = 30, Y is 30 + 12.