

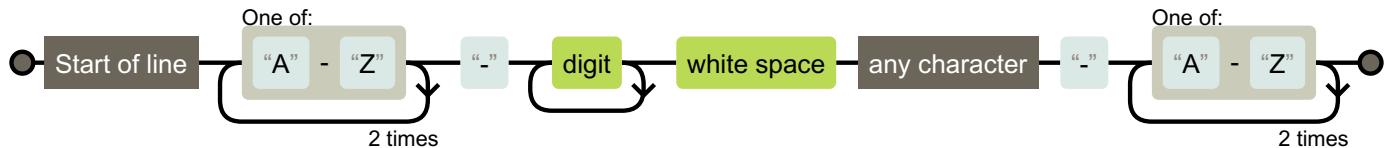


Θέμα 1 [A = 2 μονάδες, B = 1 μονάδα]

A) Δίνεται ο παρακάτω κώδικας σε C που υποθετικά βρίσκεται σε ένα αρχείο με όνομα exams.c	B) Δίνεται ο παρακάτω κώδικας σε C:
#include <stdio.h> void fun(int a, int b[], int n) { a = 1; b[0] = 1; } int main(void) { int a = 0; int b[3] = {0, 0, 0}; fun(a, b, 3); printf("%d\n", a); printf("%d %d %d\n", b[0], b[1], b[2]); }	#include <stdio.h> int a = 0; extern int b; void increment() { static int c = 0; int a = 0; a++; b++; c++; printf("a = %d, b = %d, c = %d\n", a, b, c); } int main() { increment(); increment(); increment(); return 0; } int b = 0;
I. Γράψτε τις εντολές μεταγλώπτισης και εκτέλεσης του παραπάνω κώδικα καθώς και ποιο θα είναι το αποτέλεσμα της εκτέλεσης του. II. Γιατί ο κώδικας αυτός αποτελεί ένα παράδειγμα παραβίασης της ορθογωνικότητας από τη γλώσσα C;	I. Τι θα εμφανίσει κατά την εκτέλεσή του ο παραπάνω κώδικας; II. Αν αφαιρεθεί η εντολή int a=0; μέσα από τη συνάρτηση increment, θα προκύψει σφάλμα μεταγλώπτισης ή όχι και γιατί;

Θέμα 2 [2 μονάδες]

Δίνεται σχηματικά η ακόλουθη κανονική έκφραση:



Δίνεται ότι:

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

Συμπληρώστε τα 5 κενά στον ακόλουθο κώδικα Python έτσι ώστε να εκτυπώνονται όλες οι εμφανίσεις του προτύπου που ορίζει η παραπάνω κανονική έκφραση στο κείμενο text:

```
import _____ # (1)
text = """ABC-123 .-XYZ
DEF-46E A-XYZ
GHI-789 K-AB1"""
pattern = r"_____ " # (2)
matches = re.findall(pattern, _____, re.MULTILINE) # (3)
for match in _____: # (4)
    print(_____) # (5)
```

Θέμα 3 [Α = 1 μονάδα, Β = 2 μονάδες]

Α) Δίνονται σε Python οι συναρτήσεις my_gcd, my_lcm για την εύρεση του Μέγιστου Κοινού Διαιρέτη και του Ελάχιστου Κοινού Πολλαπλασίου δύο ακεραίων αντίστοιχα καθώς και κώδικας με unit tests για τον έλεγχο των συναρτήσεων (με κάποια κενά). Συμπληρώστε τα κενά στον ακόλουθο κώδικα έτσι ώστε ο κώδικας να λειτουργεί και να «περνάνε» όλα τα unit tests.

```
import _____ # 1
def my_gcd(a, b):
    if b == 0:
        return a
    else:
        return my_gcd(b, a % b)
def my_lcm(a, b):
    if a == 0 or b == 0:
        return 0
    return a * b // gcd(a, b)
class TestMathFunctions(______): # 2
    def test_gcd(self):
        self.assertEqual(gcd(48, 18), _____) # 3
    def test_lcm(self):
        self.assertEqual(lcm(4, 6), _____) # 4
if _____ == '__main__': # 5
    unittest.main()
```

Β) Γράψτε ισοδύναμο κώδικα σε Haskell για τις συναρτήσεις my_gcd και my_lcm (που δίνονται παραπάνω σε Python). Περιγράψτε τη διαδικασία με την οποία μπορούν να κληθούν οι δύο αυτές συναρτήσεις μέσα από το ghci της Haskell για τα ζεύγη τιμών 48, 18 και 4, 6 αντίστοιχα.

Θέμα 4 [2 μονάδες]

Δίνεται η ακόλουθη βάση γνώσης στην Prolog:

```
movie('Inception', 'Nolan', 2010).
movie('Interstellar', 'Nolan', 2014).
movie('Pulp Fiction', 'Tarantino', 1994).
movie('Django Unchained', 'Tarantino', 2012).
movie('The Matrix', 'Wachowski', 1999).
```

Γράψτε τα Prolog κατηγορήματα:

- I. **directed_by(Director, Title)**, που να επιστρέφει τους τίτλους ταινιών ενός σκηνοθέτη.
- II. **before_year(Year, Title)**, που να επιστρέφει τους τίτλους ταινιών που έχουν κυκλοφορήσει πριν από μια ημερομηνία.
- III. **directed_by_before_year(Director, Year, Title)**, που να επιστρέφει τους τίτλους ταινιών ενός σκηνοθέτη που έχουν κυκλοφορήσει πριν από μια ημερομηνία.

Επιπλέον, θέστε το ερώτημα:

- IV. Ποιες είναι οι ταινίες που έχουν σκηνοθετηθεί από τον σκηνοθέτη Tarantino πριν το 2000;