



Θέμα 1 [2 μονάδες]

Δίνεται η ακόλουθη γραμματική:

```
<assign> -> <id> = <expr>
<id> -> X | Y | Z
<expr> -> <expr> - <term> | <term>
<term> -> <term> / <factor> | <factor>
<factor> -> ( <expr> ) | <id>
```

Γράψτε μια αριστερότερη παραγωγή (leftmost derivation) και σχεδιάστε το αντίστοιχο δένδρο συντακτικής ανάλυσης (parse tree) για την πρόταση: $X = (Y - X) / Z$

Θέμα 2 [A = 1 μονάδα, B = 2 μονάδες]

A. Συμπληρώστε τον ακόλουθο κώδικα σε C++ έτσι ώστε κατά την εκτέλεσή του να εμφανίζει την τιμή 42. Συμπληρώστε κώδικα μόνο εκεί όπου βρίσκονται τρεις τελείες.

```
#include <iostream>

void bar(...) {
    ...
}

int main() {
    int x = 0;
    bar(...);
    std::cout << x << std::endl;
}
```

B. Στην Python ορίστε την κλάση Weight έτσι ώστε να δημιουργούνται αντικείμενα βαρών σε κιλά ή σε λίβρες με εντολές της μορφής:

```
Weight(3.5, "KG")
Weight(3.5, "LB")
```

Ορίστε τη μέθοδο `__str__` της Weight. Υπερφορτώστε τον τελεστή `+` έτσι ώστε να αθροίζονται δύο αντικείμενα Weight και να προκύπτει ένα νέο αντικείμενο. Το αποτέλεσμα να είναι στις μονάδες του αριστερού τελεστέου της πράξης. Δίνεται ότι 1 κιλό ισούται με 2.2046 λίβρες. Συμπληρώστε τον ακόλουθο κώδικα στη θέση του `pass` έτσι ώστε να εμφανίζει αυτά που φαίνονται δεξιά:

```
class Weight:
    pass

if __name__ == "__main__":
    w1 = Weight(10, "KG")
    print(w1)
    w2 = Weight(5, "LB")
    print(w2)
    print(w1 + w1)
    print(w1 + w2)
    print(w2 + w1)
```

Κώδικας (να συμπληρωθεί)

```
10.0000 KG
5.0000 LB
20.0000 KG
12.2680 KG
27.0460 LB
```

Επιθυμητή έξοδος

Θέμα 3 [A = 1 μονάδα, B = 1 μονάδα]

A. Ο ακόλουθος κώδικας είναι σε Python και υλοποιεί ένα γραφικό περιβάλλον διεπαφής με τη βιβλιοθήκη `tkinter`. Σχεδιάστε το τι θα εμφανίσει κατά την εκτέλεσή του και περιγράψτε ποια θα είναι η λειτουργικότητά του.

```
import tkinter as tk

root = tk.Tk()
v1 = tk.StringVar()
```

B. Δίνεται η ακόλουθη λίστα φρούτων:

```
fruits = ['mango', 'kiwi', 'strawberry', 'guava', 'pineapple', 'mandarin orange']
```

Γράψτε περιφραστικές λίστες (comprehensions) σε Python για τα ακόλουθα:

1. Λίστα με όλα τα φρούτα με κεφαλαία (χρησιμοποιήστε τη μέθοδο των λεκτικών `upper()`).

```

v1.set("0.0")
v2 = tk.StringVar()
v2.set("0.0")

def foo(*args):
    if root.focus_get() != v1_entry:
        return
    try:
        v1_cm = v1.get()
        v2.set(f"{float(v1_cm)*2.54:.2f}")
    except ValueError as e:
        print(e)

def bar(*args):
    if root.focus_get() != v2_entry:
        return
    try:
        v2_inch = v2.get()
        v1.set(f"{float(v2_inch)/2.54:.2f}")
    except ValueError as e:
        print(e)

root.geometry("350x300")
root.title("EXERCISE")

v1.trace_add("write", foo)
v2.trace_add("write", bar)
v1_entry = tk.Entry(root, width=10,
textvariable=v1)
v1_entry.grid(row=0, column=0)
label1 = tk.Label(root, text="CM = ")
label1.grid(row=0, column=1)
v2_entry = tk.Entry(root, width=10,
textvariable=v2)
v2_entry.grid(row=0, column=2)
label2 = tk.Label(root, text="INCHES")
label2.grid(row=0, column=3)

root.eval("tk::PlaceWindow . center")
root.mainloop()

```

2. Λίστα με τα μήκη ονόματος κάθε φρούτου (χρησιμοποιήστε τη συνάρτηση που επιστρέφει το μήκος ενός λεκτικού len()).
3. Λίστα με όλα τα φρούτα με ονόματα με 5 ή περισσότερους χαρακτήρες.
4. Λίστα με όλα τα φρούτα που η ονομασία τους περιέχει τον χαρακτήρα r (μπορείτε να ελέγξετε αν ένα λεκτικό βρίσκεται σε ένα άλλο λεκτικό με τον τελεστή in, για παράδειγμα 'h' in 'hello' επιστρέφει True).

Θέμα 4 [A = 1 μονάδα, B = 2 μονάδες]

A. Ποια θα είναι τα αποτελέσματα των ακόλουθων εντολών της Haskell αν δοθούν στο ghci;

1. ghci> (\x y -> (x + y)/2) 5 7
2. ghci> [0, 5 .. 25]
3. ghci> take 4 [100, 90 ..]
4. ghci> filter odd [1 ..10]
5. ghci> 3 * (10 `div` 3) + 10 `mod` 3

B. Δίνεται ο ακόλουθος ορισμός για το κατηγορήμα member/2 στην Prolog:

```

member(X, [X|_]).
member(X, [_|T]) :- member(X,T).

```

1. Περιγράψτε τη λειτουργία του κατηγορήματος member/2.
2. Δώστε 2 παραδείγματα όπου η χρήση του κατηγορήματος member/2 γίνεται με διαφορετικό τρόπο.
3. Εξηγήστε την υλοποίηση του member/2, αναλύοντας τις 2 προτάσεις από τις οποίες αποτελείται.