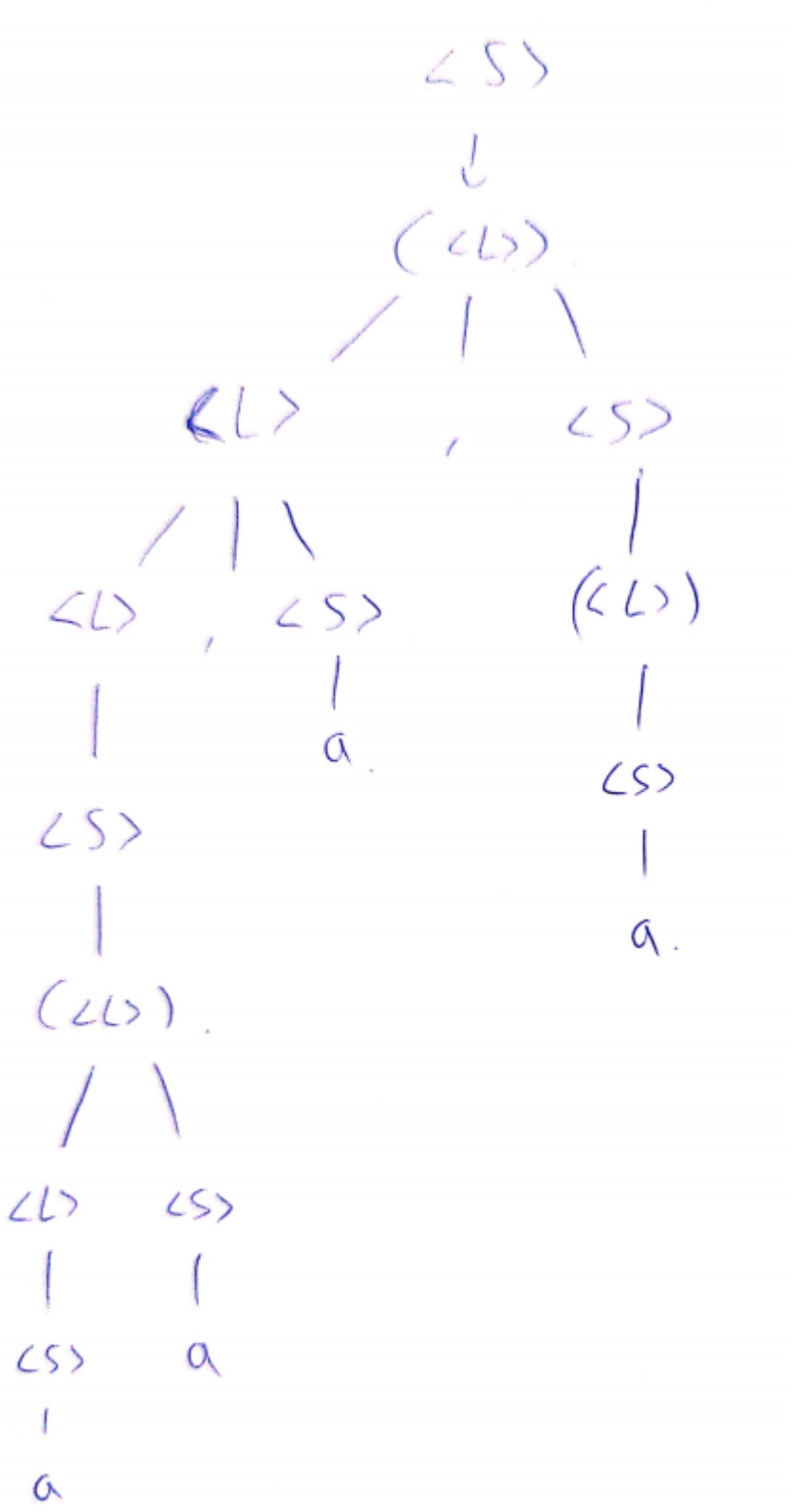
el18153 – Γιώργος Κυριακόπουλος

1.
α)



β) όχι δεν είναι, δεν βρήκαμε δυο δέντρα για κάποια έκφραση.
γ) α ή παρενθέσεις με :α ή παρενθέσεις από διάφορο πλήθος ά μέσα.

2.
β) Με εκτέλεση unique([42, 42]) βγάζει false. Η ελάχιστη αλλαγή που μπορούμε να κάνουμε είναι να βάλουμε ! ανάμεσα στο member(Item, Rest) και του fail, ώστε να σταματάει στην πρώτη ενοποίηση, δηλαδή member(Item, Rest), !, fail.

γ1) 5 17 3 42 5 17
γ2) 5 17 3 42 42
δ1) 5 3 5 1
δ2) 5 5 1 1

5.
a)
```
maxd(A , A) :- integer(A).

maxd(n(A, B, C) , Max) :- maxd(A, X), maxd(B, Y), maxd(C, Z), Mid is max(X, Y), Max is max(Mid, Z).

replacemaxd(A, Max, Max):- integer(A).

replacemaxd(n(A, B, C), Max, n(FinalA, FinalB, FinalC)) :- replacemaxd(A, Max, FinalA), replacemaxd(B, Max, FinalB), replacemaxd(C, Max, FinalC).

maximize(A, MaxTree) :- maxd(A, Max), replacemaxd(A, Max, MaxTree).
```

β)
```
oddhelper(A, A) :- integer(A).

oddhelper(n(A, B, C), n(Anew, Bnew, Cnew)) :-
    oddhelper(A, Anew),
    oddhelper(B, Bnew),
    oddhelper(C, Cnew),
    (checkn(Anew); checkn(Bnew); checkn(Cnew)).


oddhelper(n(A, B, C), Term) :-
    oddhelper(A, Anew),
    oddhelper(B, Bnew),
    oddhelper(C, Cnew),
    integer(Anew),
    integer(Bnew),
    integer(Cnew),
    Sum is Anew + Bnew + Cnew,
    (mod(Sum, 2) =:= 0 ->  n(Anew, Bnew, Cnew) = Term ; Term is 17).

unoddsum(A, Fin) :- once(oddhelper(A, Fin)).
```

checkn(n(, , _)).

6.

```
def sliding (a, k):
        n = len(a)
        seen = {}
        for i in range(n – k + 1):
                sum = 0
                for j in range (i, i + k):
                        sum += a[j]
                if sum in seem:
                        seen[sum] += 1
                else:
                        seen[sum] = 1

max_value = max(seen.values())
max_key = 0

for i in seen:
        if i > max_key and seen[i] = max_value:
                max_key = I
print(max_key, max_value)
```

Χρονική πολυπλοκότητα O(n^2) αφού έχουμε δύο εμφωλευμένες for και το in είναι O(1).