

BIRKBECK, UNIVERSITY OF LONDON

Fundamentals of Computing

Coursework 2

BARAN BULUTTEKIN

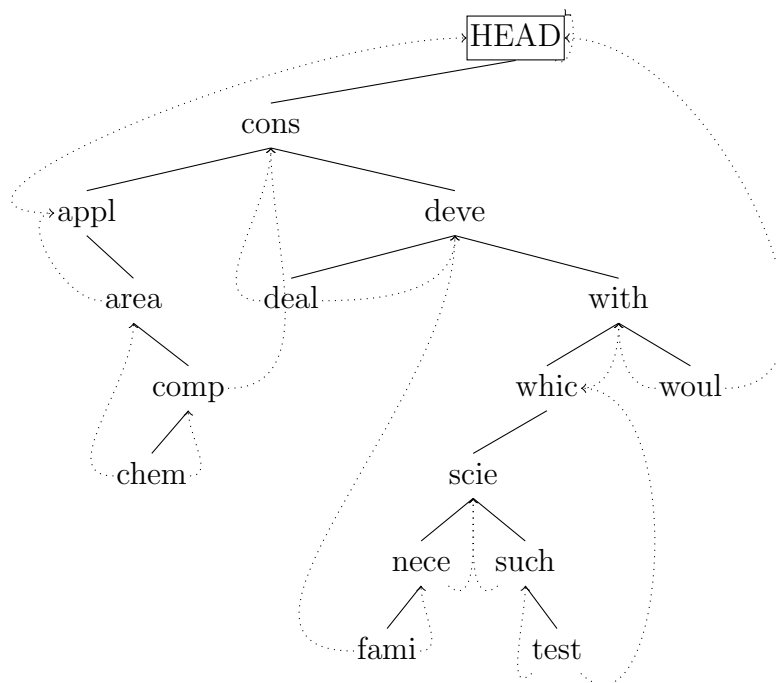
13153116

I have read and understood the sections of plagiarism in the College Policy on assessment offences and confirm that the work is my own, with the work of others clearly acknowledged. I give my permission to submit my report to the plagiarism testing database that the College is using and test it using plagiarism detection software, search engines or meta-searching software.

Answers

- Baran Buluttekın
- Summation of values = 171
- Line number = 72
- X = 1 its 5th word
- words = consisted developing application area with which computer scientist would necessarily familiar such chemical testing dealing
- four letter version = cons, deve, appl, area, with, whic, comp, scie, woul, nece, fami, such, chem, test, deal

1. Threaded binary tree:



2. *Post-order* traversed: chem, comp, area, appl, deal, fami, nece, test, such, scie, whic, woul, with, deve, cons
3. Traversed *pre-order* with algorithm from p.16: cons, appl, area, comp, chem, deve, deal, with, whic, scie, nece, fami, such, test, woul

For the stack below left most item represent the first item in and right most item is the last item get in to the stack. Words crossed out represent item that was in the stack but popped out.

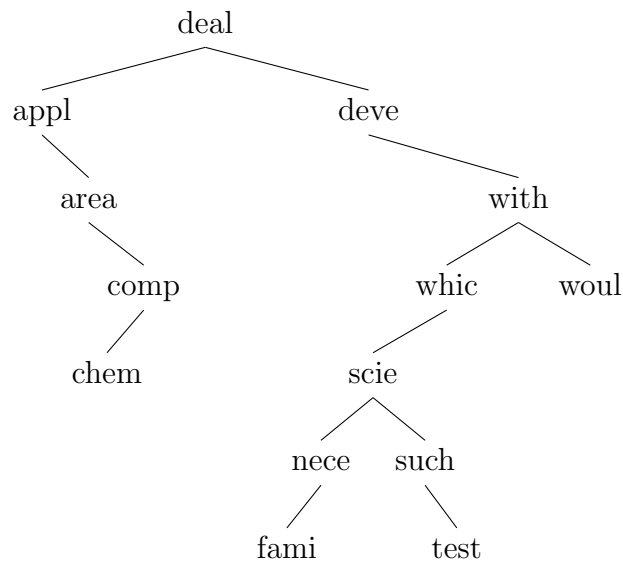
My 3th node is appl,
 Nodes visited: cons, appl
 Stack: cons

My 6th node is whic,
 Nodes visited: cons, appl, area, comp, chem, deve, deal, with, whic
 Stack: ~~cons~~, ~~appl~~, area, ~~comp~~, ~~chem~~, deve, deal, with

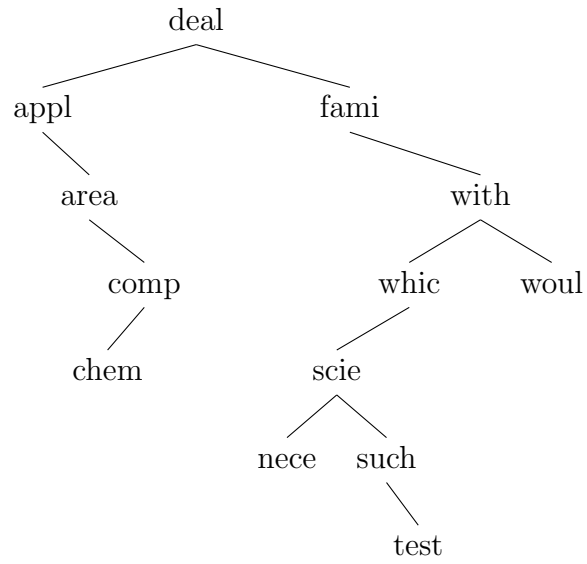
My 9th node is woul,
 Nodes visited: cons, appl, area, comp, chem, deve, deal, with, whic, scie, nece, fami, such, test, woul
 Stack: ~~comp~~, ~~appl~~, ~~area~~, ~~comp~~, ~~chem~~, ~~deve~~, ~~deal~~, ~~with~~, ~~whic~~, ~~scie~~, ~~nece~~, ~~fami~~, ~~such~~, test

My 12th node is such,
 Nodes visited: cons, appl, area, comp, chem, deve, deal, with, whic, scie, nece, fami, such
 Stack: ~~cons~~, ~~appl~~, ~~area~~, ~~comp~~, ~~chem~~, ~~deve~~, ~~deal~~, ~~with~~, ~~whic~~, ~~scie~~, ~~nece~~, fami

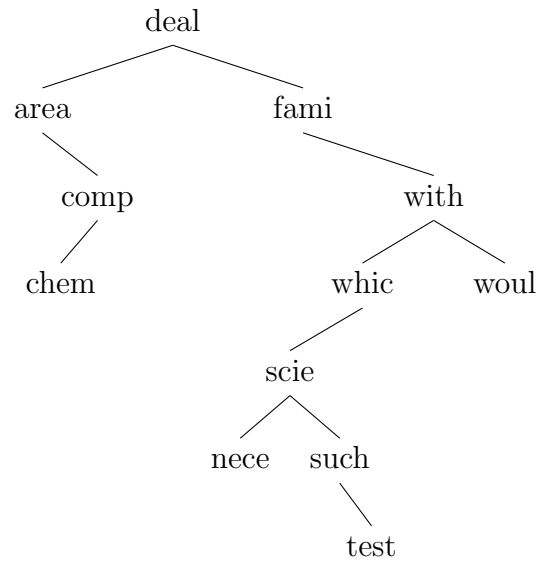
4. 1st item is cons, when removed:



2nd item is deve, when removed:

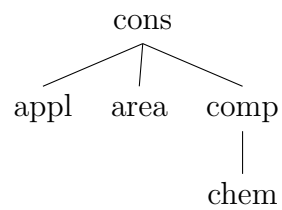


3rd item is appl, when removed:



5. We will obtain 4 following trees.

1st tree :



$2^{nd}tree :$

deve
|
deal

$3^{rd}tree :$

with
|
whic
/ | \
scie such test
|
nece
|
fami

$4^{th}tree :$

Node: woul

6. Algorithm:

```
void pre(Treenode P)
{
    visit(P);
    pre-treverse(P↑LLINK);
    pre-treverse(P↑RLINK);
}
```