# Algorithms and Data Structures
# Week 6: Finishing off trees

**Abstract**

**Summary:** This week we'll finish off the material we covered on trees. I will also spend some of the lecture time on the assignment specification.

· **Exercises to hand in** Hand in the answers to questions below. When you have finished these, you can spend the rest of the workshop getting started with the assignment.

## 1 Group exercises

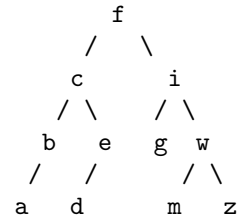
1. Arrange nodes that contain the letters `A, C, E, F, L, V` and `Z` into two binary search trees, one with maximum height and one that has minimum height.

2. Now write a recursive function which takes a binary search tree and adds the values of the nodes into a stack so that the items are sorted. You may use the standard functions of a basic stack class.

```
  void inStack(IntBSTNode t, Stack st){
//PRE:  t is a binary search tree and st is a stack of items sorted from bottom to top
//      that all the items of t are greater than the items of the stack
//POST: Adds the items of t to the stack in correct sorted order
```

   Make sure that you comment your code clearly.

3. Use the deletion function given in lectures to redraw the tree after deleting the following letters. First delete `i`, then `f`, then `a`. Draw the resulting tree after each deletion.




4. Write a program that takes a binary search tree, and two items `a, b` with `a <= b` and prints out all the nodes have data lying between `a` and `b`.