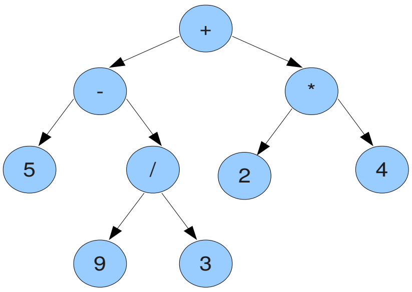
MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

159.201 Algorithms & Data Structures
Summer 2014

Tutorial 5

1. A Tree class was presented in the slides. The main function can be used to manually build a tree, creating nodes from bottom-to-top (leaves to root). Modify the Tree class and the main function to create a tree like the one in the figure below:

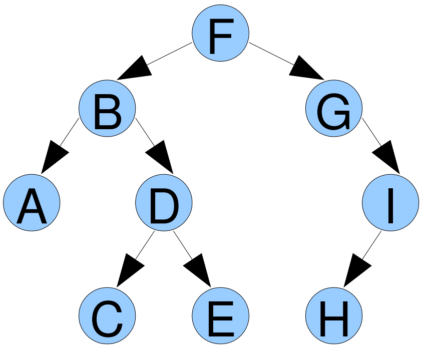


Use **`chars`** for the data type. Use the three traversal functions and check that they work correctly:
Pre-order:
In-order:
Post-order:

2. Modify the recursive ***In-order*** function to be able to fully parenthesise the in-order traversal, i.e., the in-fix expression from the Tree above.

3. The algorithms for traversing a Binary tree are easy to implement as recursive functions. Write a program that implements an ***iterative*** algorithm to perform an in-order traversal on a Binary tree.
Tip: use a ***stack*** to store temporary pointers.

4. Draw the threads in the following tree to assist a in-order traversal. How about other traversal orders?



5. Write a C++ program that modifies all the NULL pointers of an existing tree into threads.
Check your algorithm with the tree of the previous question.
(**Tip:** start with the standard InOrder() function and keep track of both the previous and next nodes).