



First test

1 Warming up

1. Draw the **Binary Search Tree** after inserting the values $[5, 2, 7, 1, 3, 4, 6, 8]$ (like in lecture 2).
2. Compute the height of the **Binary Search Tree** after inserting $[5, 2, 7, 1, 3, 4, 6, 8]$.

2 Programming part

You are asked to program a method for the iterative pre-order binary tree traversal. Here are some hints: recall that the pre-order traversal of a tree resembles the depth-first search traversal of a graph. For this you'll need to make use of a stack. The class `Stack` is provided.

1. Connect to <http://deptinfo.unice.fr/~bmartin/mathmods.html>
2. Download and expand the archive `Trees.zip` from the additional Material columns
3. Go to the directory which contains the files:
 - `BTree.rb` that you **MUST** modify;
 - `test.rb` which provides the definition of a simple tree called `T` and which calls the method `T.traverseTypes`. The method `traverseTypes` calls the recursive `preOrder` traversal (which is provided) and your `iterPreOrder` traversal and prints their outputs. You don't need to modify this file.
4. In the file `BTree.rb`, complete the method `iterPreOrder` which was intentionally left blank between the lines

```
# YOU INSERT YOUR CODE BELOW and
# STOP INSERTING YOUR CODE.
```

5. You can try your work within an interactive ruby session (`irb`), by launching the command:

```
load ``test.rb``
```

6. When you're done, do not forget to add your name and surname at the top of the file and send it by mail to `Bruno.Martin@unice.fr` with the subject `Mathmods Test`.

3 Additional questions

1. Explain how you could compute the tree height by an algorithm and propose an algorithm (or ruby program)