Homework 4

Part A

Klevis Tefa

Question A.1

(a) This program compiles and acts as it ought.
(b) The two odd lines in this program are lines 6 and 7. On line 6 the argument for new Node should be an Integer object however a primitive type int is passed as an argument. This process is called *autoboxing*, where the primitive type int is automatically converten to the Integer object that is passed as a parameter. In line 7 it happens the reverse process of what happens in line 6, *unboxing*. Here an Integer object is assigned to a primitive type variable. As in autoboxing the convertion from Integer object to int happens automatically.

Question A.2

(a) We can keep a counter that starts at 0 (balanced) and increment it every time there's '(' and decrement every time we encounter ')'. However if the counter becomes negative that means it's not well-formed.
(b) You can use counters but you'd have to keep track of which counters were incremented first and decremented after which would complicate our solution a lot. However, we can use an ArrayList instead of a stack and add elements everytime a parenthesis is opened and delete element every time is closed and if the ArrayList is null at the end then it means that the expression is well-formed. However, this is pretty much acts like a stack but it's implemented on an ArrayList.

Question A.3

(a) We can put any word from the list A to list B and that would make all the solution to fail. Or another word would be putting the string *p* to list A.
(b) pat = java; rep = class;
(c) 5, 5, 7, 11 tokens respectively for each string. Every time the string hits a comma or space followed by any amount of space (/s) or just a dot it separates it as a token.