Lab Goal : This lab was designed to teach you more about stacks.

Lab Description : Read in a group of symbols and check to see if the appropriate opening symbol correctly matches up with the appropriate closing symbol.
The opening symbols are "`{(<[`" and the appropriate closing symbols are "`})>]`".

You must read in and analyze each group.

If you were to read in `{[]}`, you would have a correct balance of opening and closing symbols.
If you were to read in `{[}]`, you would not have a correct balance of opening and closing symbols.

Sample Data :

```
(abc(*def)
[{}]
[
[{<()>}]
{<html[value=4]*(12)>{$x}}
[one]<two>{three}(four)
car(cdr(a)(b)))
car(cdr(a)(b))
```

Files Needed ::

SyntaxChecker.java
SyntaxCheckRunner.java

Sample Output :

```
(abc(*def) is incorrect.

[{}] is correct.

[ is incorrect.

[{<()>}] is correct.

{<html[value=4]*(12)>{$x}} is correct.

[one]<two>{three}(four) is correct.

car(cdr(a)(b))) is incorrect.

car(cdr(a)(b)) is correct.
```

algorithm help

```
while there are more values in the expression
{
  get a value from the input
  if you have an opening symbol
    push it on the stack
  else if it is a close symbol
    if the stack is not empty
      pop a value
      check for a match with the current close symbol
      else
        stop the process and mark the expression as bad
}
make sure nothing is left in the stack
```