

PROGRAMMING CHALLENGE 9: BRACKETS

Many programming languages use a variety of brackets eg `()`, `{}` and `[]` in their program statement constructs. Brackets can be nested eg `(())` and `{[]}`. A well-formed construct must contain a matched set of optionally nested brackets eg `(())`, `{[]}` and `(){}[]` are well-formed whereas `(()`, `{]` and `)(` are not.

An efficient way to check if a program construct is well-formed is to use a stack to scan the brackets once from left to right as follows: Start from an empty stack. Whenever we encounter an open bracket, push it into the stack. Whenever we encounter a close bracket, we check if it is of the same type with the item at the top of the stack. Once we have a match, we pop the topmost bracket from the stack. Only when we managed to read the last bracket and find that the stack is empty, then we know that the brackets are properly nested, and the program construct is well-formed.

The stack ADT contains character data and a top pointer, and has the following operations:

<code>Create()</code> :	initialise a new stack
<code>Push(item)</code> :	add item top of stack
<code>Pop()</code> :	remove item from top of stack
<code>Peep()</code> :	inspect the topmost item of stack
<code>isEmpty()</code> :	check if stack is empty

Task 1

Using object-oriented programming techniques, implement the stack ADT using the class name `Stack`.

Evidence 1: Your `Stack` class program code.

Task 2

Write a function `CheckNested` which accepts an argument of a **string** construct of nested brackets and returns a **Boolean** result indicating if construct is well-formed.

Evidence 2: Your `CheckNested` function code.

Task 3

Write code to test your `CheckNested` function using the data file provided in `DATA.TXT`. If a construct is not well-formed, append it to a text file `ERRORS.TXT`.

Evidence 3: Program code for testing `CheckNested` function.

Evidence 4: Screenshot of `ERRORS.TXT`.

Task 4

Write a function `CheckWellformed` which is modified from `CheckNested`. When a first mismatch of bracket occurs, function will display a diagnostic message "Expecting '<?>'", where ? is the type of bracket expected. For example, for the construct `()`, the diagnostic message "Expecting '({'" is displayed.

Evidence 5: Your `CheckWellformed` function code.

Task 5

Test your `CheckWellformed` function using `ERRORS.TXT`. Display the diagnostic messages to screen.

Evidence 6: Screenshot of diagnostic messages on screen.