

ESA 3

Vorraussetzungen

Um das Programm ausführen zu können wird das netcore3.1 sdk benötigt, welches hier gefunden werden kann:

- Win: <https://dotnet.microsoft.com/download/dotnet/thank-you/sdk-3.1.415-linux-x64-binaries>
- Linux: <https://dotnet.microsoft.com/download/dotnet/thank-you/sdk-3.1.415-windows-x64-installer>
- Mac: <https://dotnet.microsoft.com/download/dotnet/thank-you/sdk-3.1.415-macos-x64-installer>

Programm ausführen

Um das demo Programm auszuführen bitte folgendes Kommando im Basisordner ausführen: `ddotnet test .\GenericStackTest\GenericStackTest.csproj`

Um die Testsuite auszuführen bitte folgendes Kommando ausführen: `dotnet test .\GenericStackTest\GenericStackTest.csproj`

Zeitkomplexität

Pop()

Die Zeitkomplexität der Pop() Methode beträgt $O(1)$, da sie einen einfachen umspeichern des Pointers im `_head` Member auf den Next darstellt

Push()

Die Zeitkomplexität der Push() Methode beträgt $O(1)$, da sie einen einfachen Aufruf auf den `_head` Member darstellt

Top()

Die Zeitkomplexität der Top() Methode beträgt $O(1)$, da sie einen einfachen Aufruf auf den `_head` Member darstellt

ToString()

Die Zeitkomplexität der ToString() Methode beträgt $O(n)$, sie skaliert mit der "höhe" Stacks, da die Methode auf jedem Element einmal aufgerufen wird.

Equals()

Die Zeitkomplexität der Equals() Methode in meiner implementierung beträgt im Bestcase $O(1)$ für Stapel die sich im obersten element unterscheiden und im Worstcase $O(n)$, sie skaliert mit der "höhe" Stacks, da die Methode solange aufgerufen wird bis sich ein Element unterscheidet oder das ende des stapels erreicht ist.

Hinweis: Man könnte $O(1)$ garantieren, wenn man die gleichheit von Stapeln enger fasst als in der vorliegenden implementierung.

In der aktuellen Implementation werden werte "inhaltlich" auf gleichheit geprüft, d.h. Stapel gelten als gleich sofern alle Elemente der meinung sind sie seien gleich.

Würde man gleichheit auffassen als "muss die selbe Adresse sein" bräuchte man nur das oberste element des Stapels prüfen, ist es gleich muss auch der rest der Kette gleich sein. (Ein und das selbe objekt, kann nicht gleichzeitig auf zwei Dinge zeigen)