

Stos i notacja postfix

Miłosz Sawicki

3 kwietnia 2019

1 Założenia projektu

Program jest skonstruowany do konwertowania wyrażeń w standardowej notacji infix do notacji postfix oraz obliczania wyrażeń zapisanych w notacji postfix. Program przy wykonywaniu operacji wykorzystuje strukturę stosu, która została zaimplementowana bez użycia predefiniowanych struktur danych.

2 Opis działania programu

Program jest napisany w języku Python3. Zaimplementowany został algorytm konwersji notacji infix do postfix oraz obliczania wyrażeń w notacji postfix. Do obu funkcjonalności program używa zaimplementowanej struktury danych - stosu. Wykorzystany stos posiada funkcje push, pop, peek, isEmpty oraz size.

Program obsługuje tylko wyrażenia zapisane z użyciem spacji jako element separujący kolejne liczby/operatorsy wyrażenia.

2.1 Konwersja notacji infix do notacji postfix

Tworzony jest pusty stos na którym przechowywane będą operatory działania oraz pustą listę/tablicę wynikową. Aby uwzględnić pierwszeństwo wykonywania operatorów w funkcji został użyty pomocniczy słownik zawierający wszystkie operatory z przypisanymi liczbami od 1 do 3, gdzie 1 oznacza najwyższy priorytet operatora. Program zamienia dane wejściowe z danymi na listę ciągów znaków i w pętli for czyta ją lewej do prawej, stosując kolejne kroki algorytmu zamiany notacji infix do notacji postfix. Po zakończeniu iteracji pętli for program opóźnia stos jednocześnie dodając wszystkie elementy do listy wynikowej.

2.2 Obliczanie wyrażenia zapisanego w notacji postfix

Podobnie jak w funkcji konwertującej notację, najpierw tworzony jest pusty stos, a wprowadzane dane są dzielone na listę składającą się z osobnych ciągów znaków. W pętli for dla każdego elementu listy, program wszystkie liczby dodaje na stos, a jeśli trafi na operator to jednocześnie zapisując w pamięci usuwa dwie górne liczby ze stosu i wykonuje na nich działanie danym operatorem gdzie wynik zostaje dodany do stosu. Ostatecznym wynikiem liczenia wyrażenia w notacji postfix jest górny element stosu i jest on zwracany w ostatniej linii kodu funkcji.

3 Instrukcja dla użytkownika programu

Po uruchomieniu skryptu, program pyta użytkownika o wybór algorytmu. Wpisanie do konsoli '1' wywołuje algorytm konwersji wyrażenia w notacji infix do notacji postfix. Natomiast po wpisaniu w konsoli '2' zostanie zainicjowany algorytm obliczający wyrażenie w notacji postfix.

Następnie użytkownik jest proszony do podania pełnej nazwy pliku (wraz z rozszerzeniem) z którego zostaną odczytane dane. (np. plik.txt) Preferowane jest, żeby plik z którego mają być odczytane dane znajdował się w tym samym folderze co główny skrypt.

Wyrażenia zawarte w plikach powinny być zapisane z użyciem spacji jako element separujący kolejne liczby/operatorsy wyrażenia.

Do projektu zostały dodatkowo dołączone dwa pliki ze już gotowymi wyrażeniami do sprawdzenia działania programu: plik1.txt (wyrażenie infix do konwersji) oraz plik2.txt (wyrażenie postfix do obliczenia) jednak korzystanie z nich nie jest konieczne.

Po wprowadzeniu tych danych program automatycznie wykonuje obliczenia i podaje wynik w konsoli.

4 Wnioski

- Program ilustruje praktyczne zastosowanie stosu.
- Program pozwala zrozumieć działanie notacji postfix oraz pozwala zauważyć jej praktyczność w zapisywaniu wyrażeń matematycznych.