|  |  |  |
| --- | --- | --- |
| **Autor:**  **Hubert Kowalczyk 259550** | **Struktury Danych i**  **złożoność obliczeniowa**  **Semestr letni 2022/2023** | **Termin:**  **Wtorek NP: 17:05** |
|
|
|
| **Prowadzący:**  **Dr. Inż. Tomasz Kapłon** | **Ćwiczenie 4** | **Data wykonania ćwiczenia: 23.05.2023** |
|
|
|
| **Data oddania sprawozdania**  **06.06.2023** |
|
|
|

1. **Cel ćwiczenia**

Celem ćwiczenia było zaimplementowanie jeden z czterech algorytmów tworzenia drzew wyszukiwań. Należało również zbadać czas tworzenia drzewa oraz pokazać jego wygląd dla niewielkich instancji.

1. **Implementacja**

W ramach realizacji ćwiczenia zdecydowano się na strukturę danych BST z algorytmem równoważenia. Implementacja kodu obsługuje operacje na drzewie BST takie jak wstawianie, usuwanie i wyszukiwanie elementów. Drzewo jest przechowywane w pamięci jako struktura danych, w której każdy węzeł przechowuje wartość klucza oraz wskaźniki na swoje lewe i prawe dziecko. Węzły są wstawiane do drzewa w taki sposób, że wartości kluczy w lewym poddrzewie są mniejsze niż węzeł nadrzędny, a wartości kluczy w prawym poddrzewie są większe. Zaimplementowano również algorytm DSW do równoważenia drzewa. Jest to procedura, która spłaszcza drzewo, wykonując rotacje w lewo aż do momentu, gdy nie zostaną stworzone kolejne węzły na prawym poddrzewie, a następnie równoważy drzewo przez wykonanie rotacji w lewo w odpowiednych węzłach.

1. **Opis stanowiska wykonawczego**

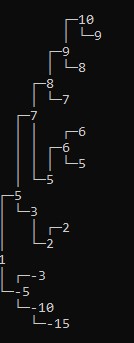
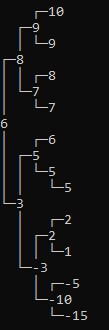
Algorytm testowano na komputerze z systemem operacyjnym Windows 10. Jednostka CPU to Intel Core i7-9750H. Sam czas wykonywania jest zapisany w pliku podanym w pliku Config.cfg. W tym pliku również można wpisać wielkość instancji, czyli liczb do dodania. Maksymalna liczba tych liczb to 1000000.

1. **Sposób przeprowadzenia badań**

Skompilowano dwa pliki wykonywalne „wersja\_testowa\_male\_instancje.exe” oraz „wersja\_badawcza.exe” Pierwszy służy do pokazania jak działa algorytm poprzez ręczne dodawanie ,usuwanie ,wyszukiwanie elementów. Pogram ten również potrafi zrównoważyć drzewo oraz pokazać jego wygląd. Program ten został użyty w celu pokazania wyglądu drzewa dla małej instancji. Drugi plik wykorzystany został do przeprowadzenia badań dla większych instancji. Pobiera on z pliku Config.cfg ścieżki do plików oraz wielkość instancji w celu przeprowadzenia badania. Zbadany zostanie czas jaki minie dodanie N elementów do drzewa, gdzie N to wielokrotności liczb 10,20,35,50,75,100 które są mniejsze od miliona. Każdą instancję przetestowano 10 razy oraz obliczono średni czas wykonywania który posłuży do dalszej analizy.

1. **Badanie algorytmu BST z równoważeniem.**

Poniżej zamieszczono przykład wyglądu drzewa przed jak i po równoważeniu dla 20 dodanych elementów

Rys 1 Drzewo przed zrównoważeniem Rys 2 Drzewo po zrównoważeniu

Poniżej przedstawiono również tabele z czasami dodawania N elementów oraz wykres na ,który zostały przeniesione te czasy.

Tabela 1 Czasy wykonywania badania w zależności od wielkości instancji

|  |  |  |  |
| --- | --- | --- | --- |
| Wielkość Instancji | Czas trwania [ms] | Wielkość Instancji | Czas trwania [ms] |
| 10 | 0,15 | 5000 | 7,94 |
| 20 | 0,13 | 7500 | 12,52 |
| 35 | 0,15 | 10000 | 17,13 |
| 50 | 0,15 | 20000 | 49,59 |
| 75 | 0,19 | 35000 | 135,26 |
| 100 | 0,24 | 50000 | 245,44 |
| 200 | 0,33 | 75000 | 541,59 |
| 350 | 0,59 | 100000 | 918,92 |
| 500 | 0,83 | 200000 | 4080,57 |
| 750 | 1,18 | 350000 | 21386,81 |
| 1000 | 1,56 | 500000 | 66083,11 |
| 2000 | 2,67 | 750000 | 194641,54 |
| 3500 | 4,96 | 1000000 | 372206,19 |

Wykres 1 Złożoność czasowa O(n) na podstawie tabeli 1