1 Lista7 Zadanie3

1.1 Implementacja na BST

Znając dokłanie jak zachowują się operacje na Drzewie Przeszukwań Binarnych, oraz ich złożónosć obliczeniową możemy użyć drzewa binarnego do stworzenia Priority Queue.

Jak wygląda Piority Queue w tym wypadku? Jako najważniejsze traktujemy ten z najmniejszym kluczem i będzie w strukturze drzewa najbardziej lewy liść. Poniżej w każdym podpukcie zaproponuję jaką funcję z BTS moża odpowiednio wykorzystać do odpowiedniej z BST oraz opisze ich złożoność.
Ważne jest to, że h oznacza w tym wypadku wysokość drzewa, natomiast n liczbę węzłów w drzewie.

1.2 Insert

Do operacji Insert, możemy wykorzystać operację Insert z BTS o złożoności

$$O(h)$$

1.2.1 Minimum

Do operacji minimum możemy wykorzystać operacje search, zaimplementowaną w taki sposób aby zwróciła wartość z najbardziej lewego liścia. Złożoność takiej operacji to

$$O(h)$$

1.2.2 ExtractMin

Do operacji extractMin możemy najpierw użyć funkcji search, zaimplementowaną w taki sposób aby zwróciła wartość z najbardziej lewego liścia. Złożoność takiej operacji to $O(h)$.
A następnie na znalezionym elemencie wykonać operację delete, która zwraca nam usunięty węzeł, jest to operacja o złożoności $O(h)$. Łączna złożoność ExtracMin wynosi $O(h) + O(h) =$

$$O(h)$$

1.2.3 DecreaseKey

Do operacj DecreaseKey możemy użyć najpierw funcji delete która zwóci nam węzeł X, złożoność ta to

$$O(h)$$

. Następnie w polu key węzła X zamienić wartość na nową. Na tak zmodywifowanym węźle X wykonujemy funkcje insert o złożoności

$$O(h)$$

Mamy tu dwie funkcje o złożoności O(h). Tak więc całkowita złożoność operacji wynosi także $O(h) + O(h) =$

$$O(h)$$

1.2.4 Delete

Do operacji Delete możemy użyć standarową operacje Delete z BTS o złożoności

$$O(h)$$

1.2.5 Union

Operacje Union możemy zrelizować poprzez wstawienie drzewa nr 2 o m węzłach do drzewa nr 1 o wysokości h. Skoro drzewo nr 2 ma m węzłów to procedure insert musimy powtórzyć m razy na drzewie nr 1. Przyjrzyjmy się najgorszemy przypadkowi. Jest to przypadek w którym, po dodaniu do drzewa nr 1, pojedynczego węzła za każdym razem zwiększamy wysokość takiego drzewa o 1. W takim wypadku złożoność wynosi.
$O(h)+(O(h+1))+O(h+2)+...+O(m-1+h) = (O(1+2+...m-1+m*h)) = (O(m^2+m*h))$
Więc koszt całej operacji w najbardziej pesymistycznym wypadku wynosi

$$O(m*h+m^2)$$

1.3 Porównanie złożoności

Operacja	Kopiec	Drzewo wysokość O(n)	Drzewo wysokość O(log(n))
Insert	O(log(n))	O(n)	O(log(n))
Minimum	O(1)	O(n)	O(log(n))
ExtractMin	O(log(n))	O(n)	O(log(n))
DecreaseKey	O(log(n))	O(n)	O(log(n))
Delete	O(log(n))	O(n)	O(log(n))
Union(z m węzłami)	O(n)	$O(m^2+m*n)$	$O(m^2+mlog(n))$