

Laboratorium AiSD

Lista 7

Wyszukiwanie, kolejki priorytetowe, tablice mieszające

Proszę pamiętać, że **część rozwiązania** zadania stanowi również **zestaw testów** zaimplementowanych algorytmów i/lub struktur danych. Dodatkowo, proszę zwracać uwagę na **powtarzające się fragmenty** kodu i wydzielać je do osobnych funkcji/klas.

1. Zdefiniuj klasę implementującą strukturę **ternarnego** (węzły z trojgiem dzieci) **kopca minimalnego** (*ang. ternary min-heap*) **`Array3Heap<T>`** zaimplementowanego **na tablicy**.

Interfejs kopca powinien udostępniać dwie operacje:

- *`void clear()`* – czyszczącą kopiec (usuającą wszystkie elementy),
- *`void add(T element)`* – wstawiającą nowy element do kopca,
- *`T minimum()`* – zwracającą minimalny element kopca wraz z jego usunięciem.

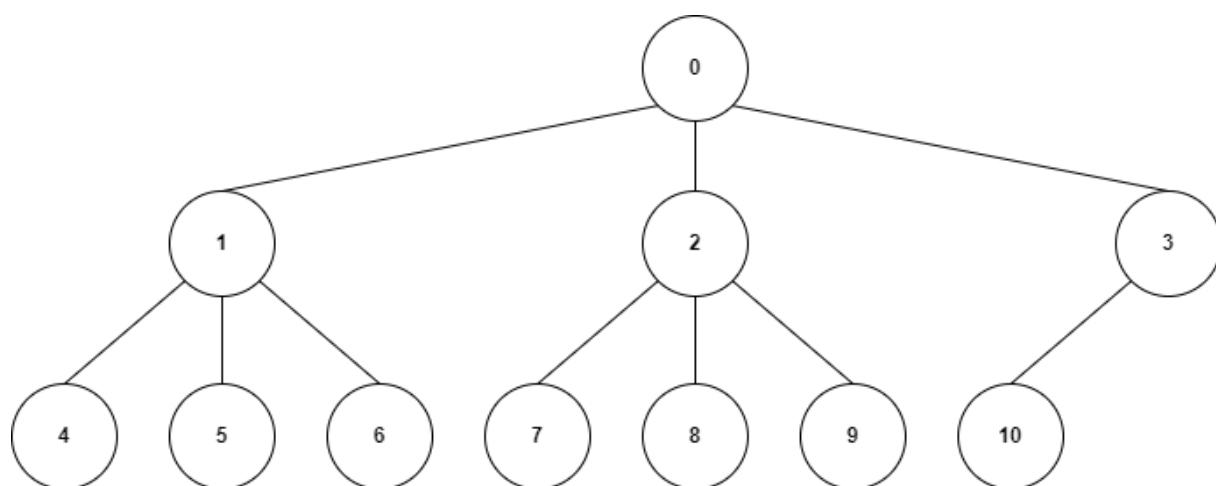
Opis budowy kopca opisano na **wykładzie 6**.

Podczas konstrukcji kopca przekaż **podstawową pojemność**, a gdy w kopcu nie ma miejsca, należy **rozmiar podwoić**.

Założ, że ***null* nie jest poprawną wartością**.

Wskazówka:

Podobnie jak w przypadku kopca binarnego, dla kopca ternarnego można wyprowadzić wzory na indeksy elementów – dzieci i rodzica – po wprowadzeniu odpowiedniej numeracji węzłów. Na podstawie rysunku takiego jak Rysunek 1, rozpisz w tabelkach pary rodzic-dziecko dla lewego, środkowego i prawego dziecka. Na ich podstawie podaj wzory na indeksy. Indeks rodzica to: $\left\lfloor \frac{i-1}{3} \right\rfloor$ – wykorzystując dzielenie całkowitoliczbowe można pominąć operację podłogi ($\lfloor \cdot \rfloor$).



Rysunek 1 Przykładowy kopiec ternarny (w węzłach podano ich indeksy!)