

# Algorytmy i struktury danych

## Lista zadań 5

### Zadanie 1

Jakie informacje przechowujemy w węźle drzewa czerwono-czarnego? Zadeklaruj strukturę `RBNode` tak, by dziedziczyła z `BSTnode`. Podaj definicję drzewa czerwono-czarnego.

### Zadanie 2

- Jaka może być minimalna, a jaka maksymalna ilość kluczy w drzewie czerwono-czarnym o ustalonej czarnej wysokości równej  $h_B$ ?
- Znajdź maksymalną i minimalną wartość stosunku ilości węzłów czerwonych do czarnych w drzewie czerwono-czarnym.

### Zadanie 3

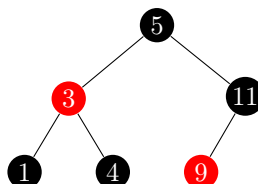
Uzasadnij posługując się rysunkiem i opisem, że operacje wykonywane w trakcie wstawiania do drzewa czerwono-czarnego (rotacja i przekolorowanie) nie zmieniają ilości czarnych węzłów, na żadnej ścieżce od korzenia do liścia.

### Zadanie 4

- Narysuj poprawne drzewo czerwono-czarne w którym na lewo od korzenia jest 1 węzeł a na prawo 7 węzłów.
- Czy istnieje poprawne drzewo czerwono-czarne, w którym na lewo od korzenia będzie 100 razy mniej węzłów niż na prawo od korzenia?

### Zadanie 5

W poniższym drzewie czerwono-czarnym:



- wstaw do niego 10.
- usuń z wyjściowego drzewa 1.

### Zadanie 6

(3 pkt.) Do pustego drzewa czerwono-czarnego wstaw kolejno 20 przypadkowych kluczy. Następnie usuń je w tej samej kolejności w jakiej wstawiałeś. Przypadkowymi kluczami są kolejne litery Twojego nazwiska, imienia i adresu. Zadanie wykonujemy na kartce (lub w pliku) i oddajemy prowadzącemu. Zadanie jest obowiązkowe.

## Zadanie 7

Analizując kod programu `RBT.h` udowodnij, że w trakcie wstawiania do drzewa czerwono-czarnego wykonają się co najwyżej dwie rotacje. Czy tak samo jest w przypadku usuwania?

## Zadanie 8

Uzasadnij, że rozmiar stosu ( $n = 100$ ) przyjęty w procedurach `insert` i `remove` w pliku `RBnpnr.h` nigdy nie okaże się za mały.