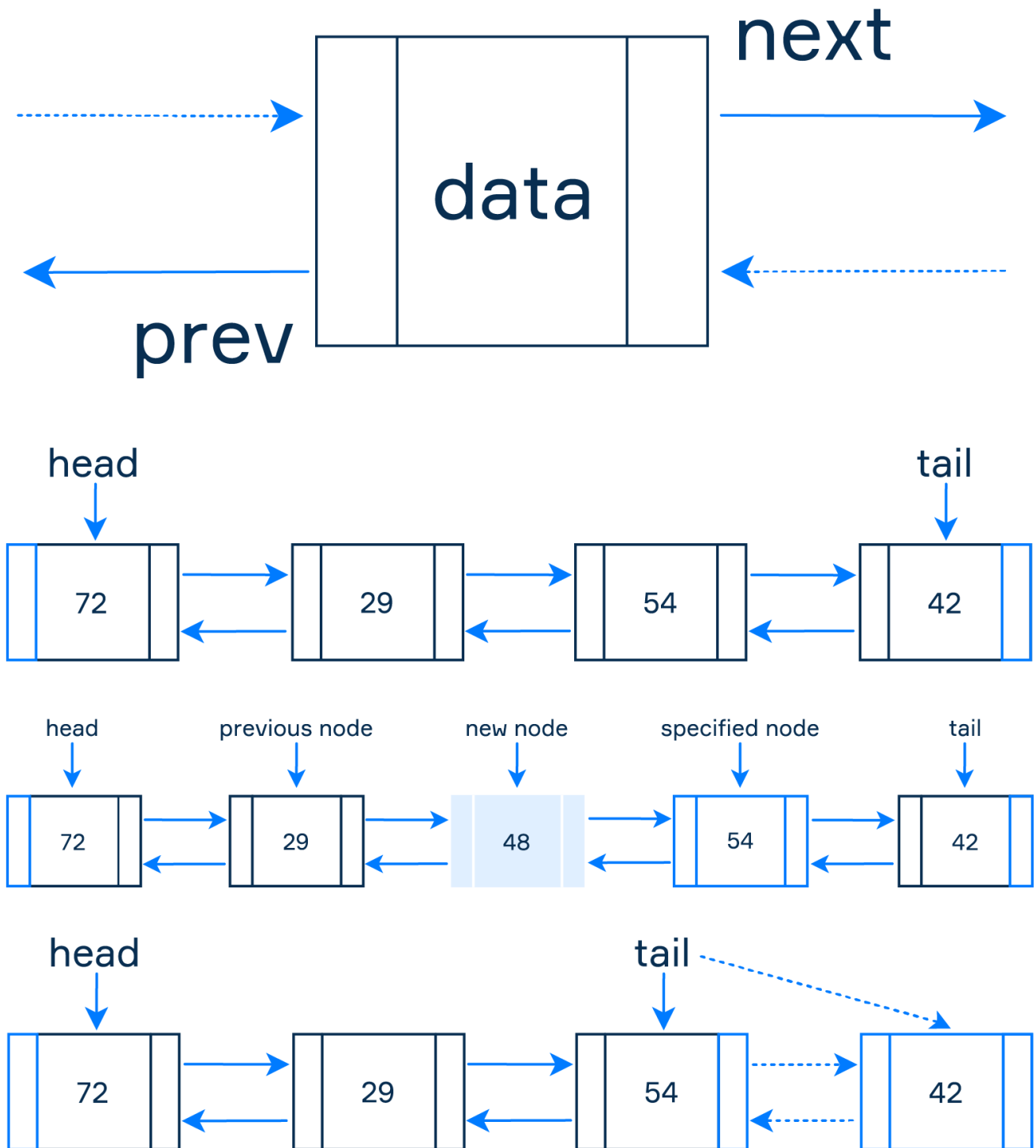


Kolekcje

Zadanie 1.

Krótki wstęp o rodzajach list. Niektóre bazują na tablicach, inne nie. Przejście do DoublyLinkedList. lista bazująca na połączeniach (DoublyLinkedList) - prezentacja struktury takiej listy i referencji pomiędzy obiektami.



DoublyLinkedList - kod:

```
static class Node<E> {  
  
    private E value;  
    private Node<E> next;  
    private Node<E> prev;  
  
    Node(E element, Node<E> next, Node<E> prev) {  
        this.value = element;  
        this.next = next;  
        this.prev = prev;  
    }  
  
    Node<E> getNext() {  
        return next;  
    }  
  
    Node<E> getPrev() {  
        return prev;  
    }  
}
```

```

class DoublyLinkedList<E>{

    private Node<E> head;
    private Node<E> tail;
    private int size;

    public DoublyLinkedList() {
        size = 0;
    }

    public Node<E> getHead() {
        return head;
    }

    public Node<E> getTail() {
        return tail;
    }

    public int size() {
        return size;
    }

    public boolean isEmpty() {
        return size == 0;
    }

    public String toString() {

        Node<E> tmp = head;
        StringBuilder result = new StringBuilder();

        while (tmp != null) {
            result.append(tmp.value).append(" ");
            tmp = tmp.next;
        }

        return result.toString();
    }
}

```

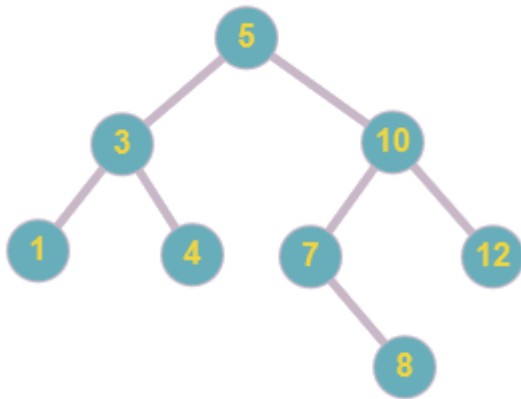
Zadanie to napisać metody dodające elementy i usuwające elementy z listy.

- addFirst(){}
- addLast(){}
- add(elem){}
- removeFirst(){}
- removeLast(){}

- remove(elem){}

Zadanie 2.

Struktura drzewa wykorzystywana w kolekcjach. opis wyglądu drzewa na przykładzie BinarySearchTree.



Opis operacji dodawania, usuwania i szukania z obrazkami.

Opis implementacji BinarySearchTree w Javie.

```
static class Node {  
    int key;  
    int value;  
    Node left;  
    Node right;  
    Node parent;  
  
    public Node(int key, int value, Node parent) {  
        this.key = key;  
        this.value = value;  
        this.parent = parent;  
    }  
}
```

Zadanie to napisać metody:

- wstawWęzeł(){}
- znajdzElement(){}
- usuńElement(){}

Zadanie 3.

mapa - nie będąca kolekcją, ale należąca do API Collection. Różne rodzaje map reprezentują pary klucz wartość.

Zadanie to odczytanie klucza szyfrującego litery z wykorzystaniem map, a następnie dokonanie odszyfrowania wiadomości.