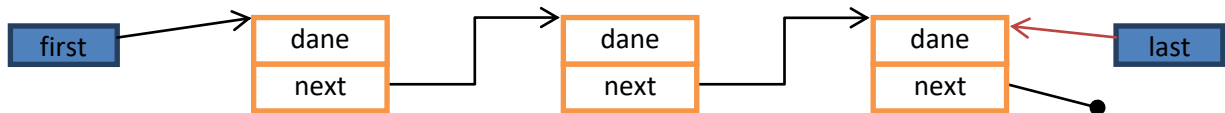


Przygotowanie do Laboratorium 4: Listy jednokierunkowe i dwukierunkowe.

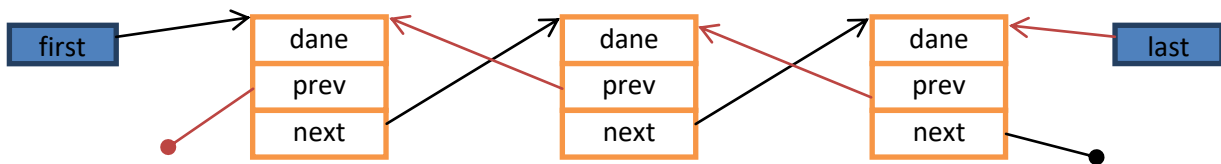
1. Koncepcja list jednokierunkowych

Lista jednokierunkowa jest strukturą danych pozwalającą na pamiętanie danych w postaci uporządkowanej, a także na bardzo szybkie wstawianie i usuwanie elementów do i z listy. Lista zapisana jest w postaci elementów zawierających porcję danych oraz wskaźnik (adres) następnego elementu. W przypadku języka Java tak naprawdę mamy do czynienia z referencjami do obiektów. W ten sposób wystarczy pamiętać referencję do pierwszego elementu listy, by pamiętać całą listę. Strukturę listy jednokierunkowej ilustruje poniższy rysunek:



2. Listy dwukierunkowe.

Listy dwukierunkowe to uporządkowane, liniowe struktury danych. Różnią się od list jednokierunkowych tylko tym, że nawigacja pomiędzy elementami listy może odbywać się w dwóch kierunkach. Typowa lista dwukierunkowa składa się z elementów, które oprócz danych przechowywanych w liście zawierają informacje o poprzednim i następnym elemencie listy. W zależności od języka programowania używanego do implementacji list odwołania do sąsiednich elementów to zazwyczaj wskaźniki lub referencje. Strukturę listy dwukierunkowej ilustruje poniższy rysunek:



3. Klasy generyczne.

Klasy generyczne są to klasy o parametryzowanych typach danych. Klasy generyczne posiadają kompletną implementację, jednak nie definiują typów danych wykorzystanych w tej implementacji. Np.:

```
public class Element<T> {  
    private T dane;  
    private Element<T> next, prev;  
  
    . . .  
}
```

Parametrem klasy Element jest typ T. Możemy teraz używać instancji tej klasy do przechowywania danych różnego typu. Np. (po zdefiniowaniu odpowiedniego konstruktora):

```
Element<Integer> e = new Element<Integer>(5);  
Element<String> e2 = new Element<String>("Ala ma kota");
```

Od wersji 7 kompilator języka Java posiada mechanizm dedukcji typów generycznych (*type inference*), dzięki czemu nie musimy za każdym razem powtarzać parametrów (typów). W związku z czym następujące wyrażenie jest poprawne:

```
Element<Integer> n = new Element<>(5);
```

Należy wykorzystać mechanizmy programowania generycznego dostępne w języku Java. **Uwaga!** Typy podawane jako parametry muszą być typami **referencyjnymi!**

4. Zadania.

1. Zaimplementować listę jednokierunkową w oparciu o strukturę wiązaną (dane przechowywane w obiektach typu Element z referencją do następnego elementu, podobnie jak dla stosu). Lista powinna umożliwiać dodawanie elementów na początek i koniec, usuwanie pierwszego i ostatniego elementu oraz wypisanie listy.
2. Zaimplementować listę dwukierunkową umożliwiającą takie same operacje jak w zad. 1.
3. W klasie reprezentującej listę zaimplementować metodę *iterator()* zwracającą obiekt implementujący `Iterator<T>`. <https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html>