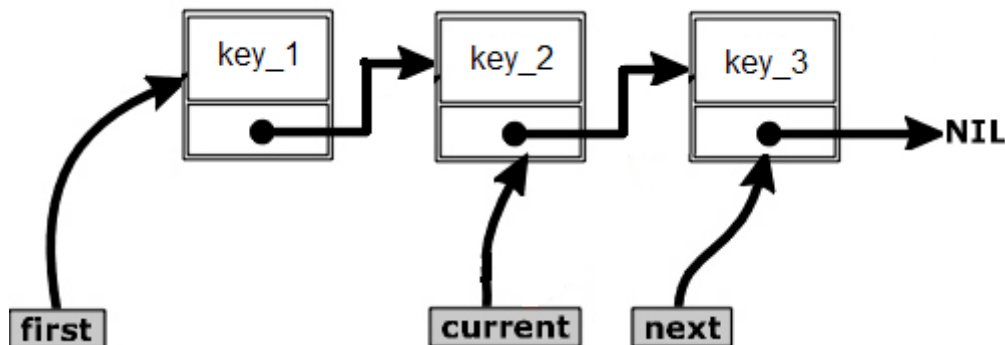


Lista jednokierunkowa

Lista jest liniowo uporządkowaną strukturą zawierającą zbiór elementów, z których dowolny element można usunąć oraz dodać w dowolnym miejscu. Pierwszy i ostatni element listy nazywamy *końcami listy*. Każdy element ma co najwyżej jednego następnika, który składa się z następujących składowych: klucza identyfikacyjnego, wskaźnika do następnika oraz dodatkowo dalszych informacji. Listy mogą być posortowane w porządku niemalejącym (w korzeniu znajduje się element o najmniejszej wartości klucza) lub nierosnącym (w korzeniu znajduje się element o największej wartości klucza). Rozważmy przypadek **jednokierunkowej listy posortowanej** w porządku niemalejącym. Aby dodać nowy element należy sprawdzić, w którym miejscu powinien się on znajdować. Dokonywane jest odpowiednie sprawdzenie rozpoczynające się w korzeniu i przechodzenie przez kolejne elementy w liście, tak długo dopóki nowy element, który chcemy dodać jest większy od badanego węzła i mniejszy od jego następnika, wtedy należy umieścić go między nimi. Należy, więc ustawić wskaźnik aktualnego węzła na dodawany element, a wskaźnik tego elementu na następnik. Ponieważ jest to lista jednokierunkowa, przeszukiwanie jej należy zawsze zaczynać od korzenia. Dodając, więc pierwszy element do pustej listy należy zapamiętać jego wskaźnik, by później mieć dostęp do tej struktury dzięki niemu. Listy mogą zawierać dwa wskaźniki. Takie listy nazywamy **dwukierunkowymi**, które pozwalają na poruszanie się w niej w obydwu kierunkach, co przyspiesza wszystkie operacje.

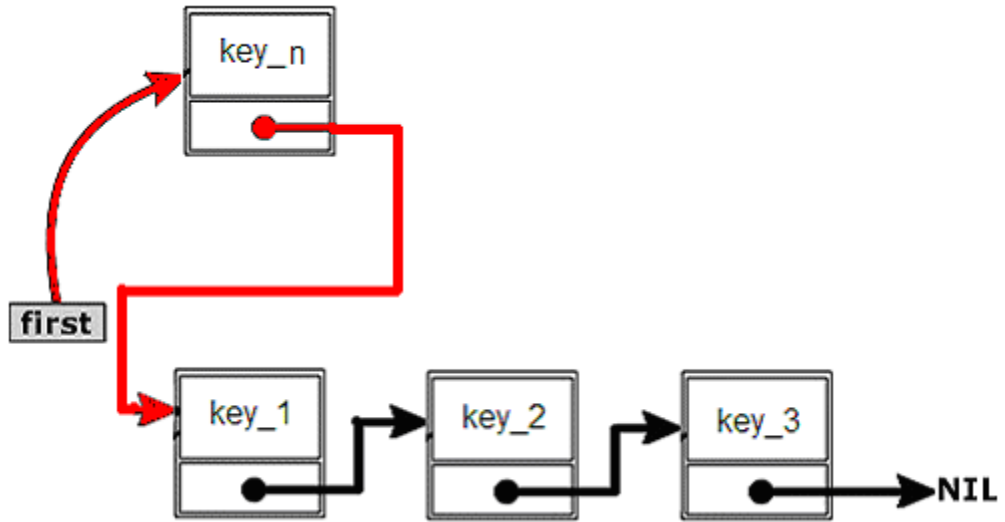
- o *Jednokierunkowa*



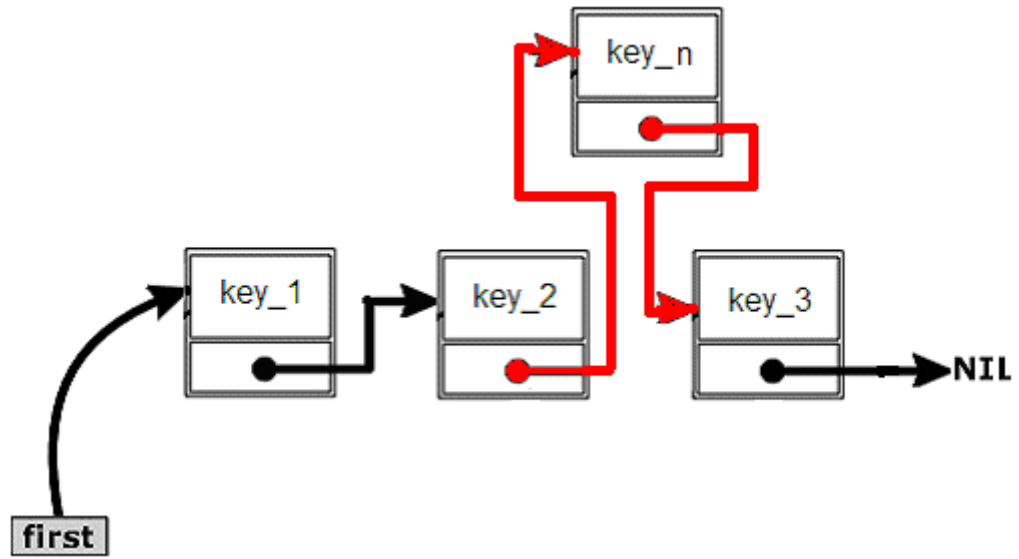
Przeszukiwanie dokonuje się od pierwszego elementu na liście (wskazywanego przez wskaźnik *first*) do ostatniego, który w polu swojego następnika wskazuje na wartość pustą (*NIL*).

Dodawanie nowych elementów

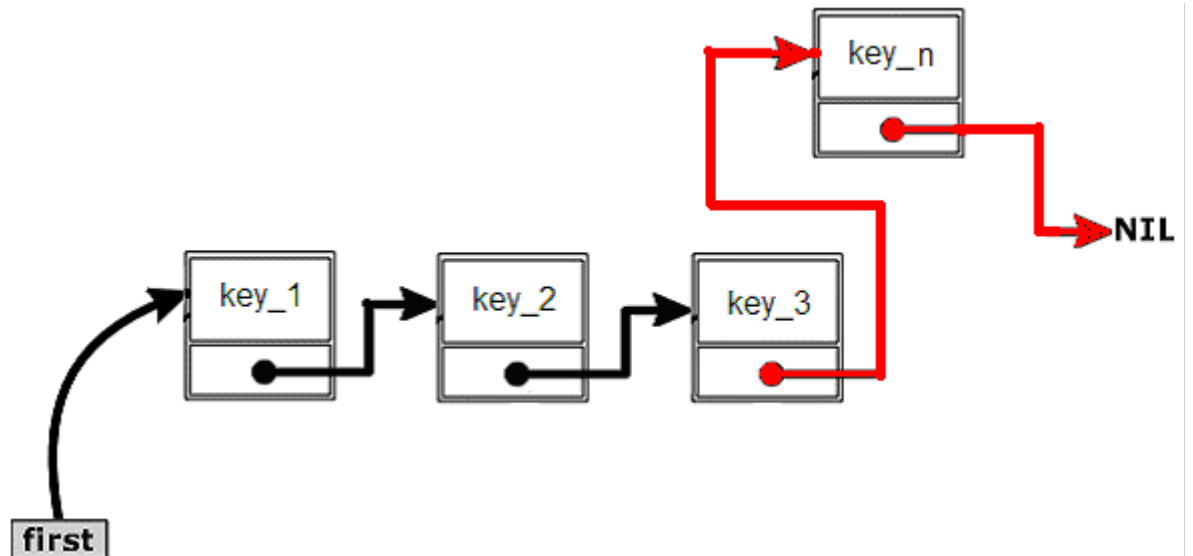
- a) na początku



b) w środku (między 2 a 3 elementem)

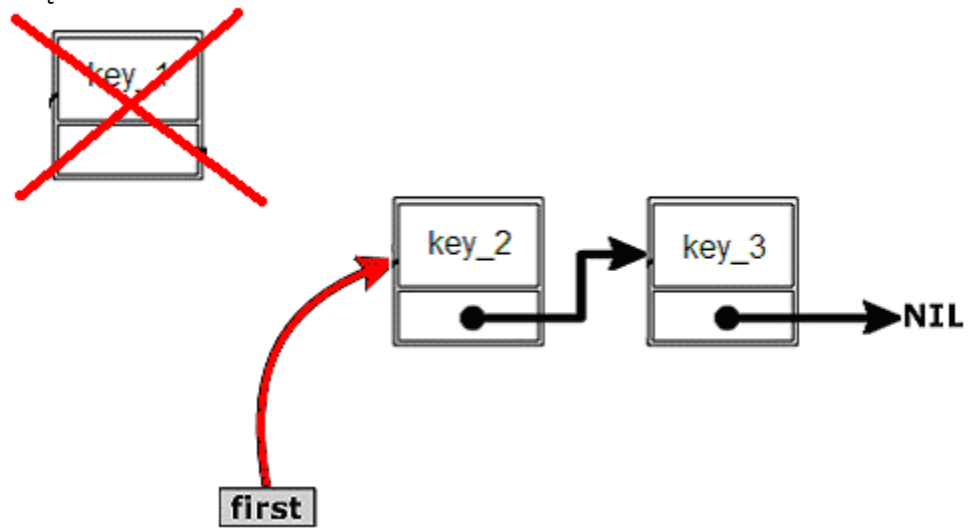


c) na końcu

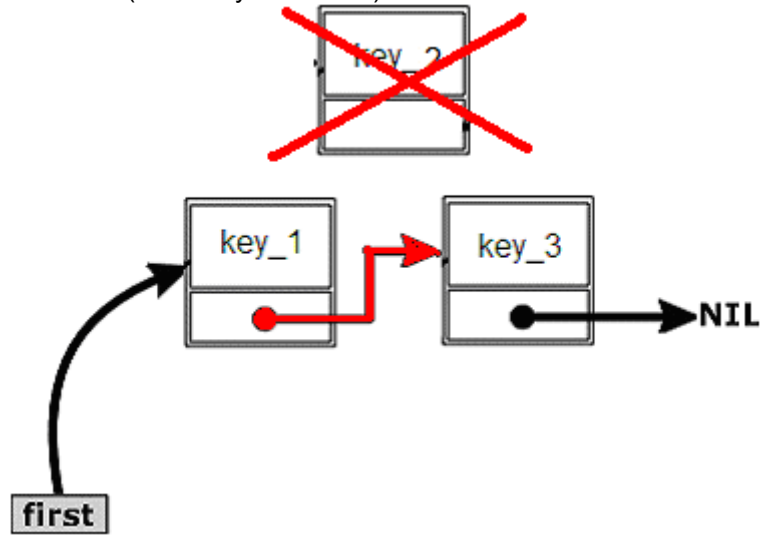


Usuwanie istniejących elementów:

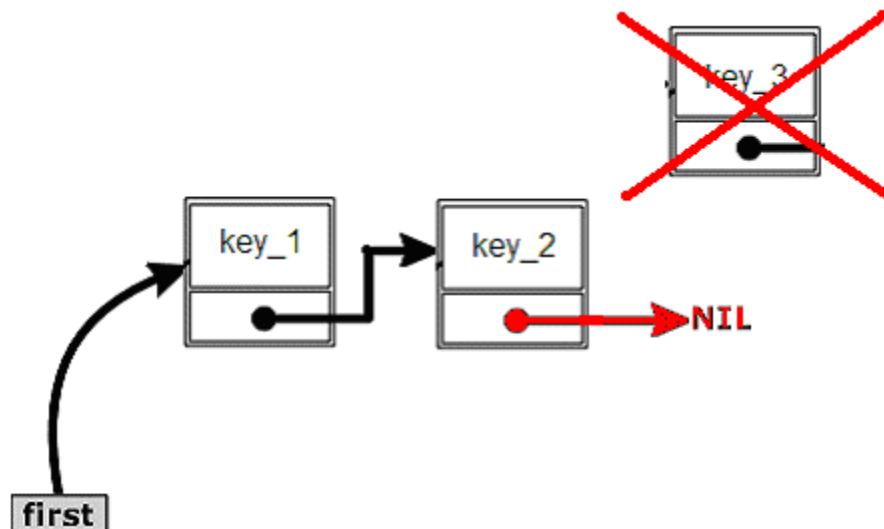
a) z początku



b) ze środka (usuwamy 2 element)



c) z końca



Przykładowa implementacja rekordu listy jednokierunkowej

```
type
  plist = ^element;
  element = record
    key : integer;
    next : plist;
  end;
```
