

Algorytmy i struktury danych

Laboratorium - lista 1

Termin wysłania do SVN: 2024-03-11

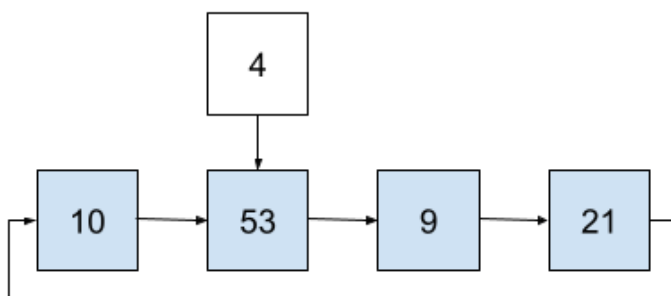
Zadanie 1. [14 p.]

Zaimplementuj kolejkę (**FIFO** first in first out) oraz stos (**LIFO** last in first out). Do każdej z nich dodaj po 50 elementów, a następnie pobierz je z kolejek (wypisuj po kolei elementy dodawane oraz wyjmowane). Pamiętaj o obsłudze błędów (próba pobrania elementu z pustej kolejki).

Zadanie 2. [15 p.]

Zaimplementuj listę **jednokierunkową cykliczną** liczb całkowitych.

Lista jest dostępna przez strukturę zawierającą liczbę elementów i wskaźnik do jednego z nich oraz liczbę elementów listy. Każdy element listy ma wskaźnik na następny element w cyklu.



- Zaimplementuj funkcje:
 - `insert(l, i)` wstawiającą nowy element z liczbą `i` do listy `l`,
 - `merge(l1, l2)` łączącą dwie listy `l1, l2` i zademonstruj jej działanie dla list długości 10 zawierających dwucyfrowe liczby nieujemne.
- Utwórz tablicę `T` zawierającą 10000 losowych liczb całkowitych z przedziału $I = [0, \dots, 100000]$, a następnie wstaw te liczby do listy `L`. Następnie wyznacz średni koszt tysiąca wyszukiwań losowych liczb z przedziału `I` na liście. (Przez "koszt" rozumiemy ilość wykonywanych porównań między elementem wyszukiwanym a elementami na liście.) Zbadaj dwa przypadki:
 - wyszukiwanie liczb, które są na liście (wybieranych losowo z tablicy `T`),
 - wyszukiwanie losowej liczby z `I`.

Zadanie 3. [15 p.]

Zaimplementuj listę **dwukierunkową cykliczną** i wykonaj te same polecenia co w poprzednim zadaniu. W tym przypadku, każdy element ma dodatkowo wskaźnik na poprzedni element w cyklu. Przy badaniu kosztów losowych wyszukiwań, każde wyszukiwanie na początku losowo decyduje w którym kierunku będzie się odbywać.