

Naloga 1

STRUKTURA

DINAMIČNI SEZNAM

Uvod

- Vaše rešitve pišite v predlogo za C++ ali C#.
- V predlogi **morajo** podani prototipi imenskih prostorov in metod ostati nespremenjeni.
- Definirate lahko poljubno število lastnih metod.

Seznam in struktura

- Podatkovna struktura *seznam*
 - [list \(C++\)](#)
 - [LinkedList \(C#\)](#)
- Struktura *Student*
 - ime (niz znakov)
 - priimek (niz znakov)
 - IDUM (niz znakov)

Implementirajte

- Delovanje metod naj ustreza definiciji dinamičnega seznama, kot ste jo spoznali na predavanjih!
- `vstavi()`
 - Vstavi podatek v obstoječe vozlišče seznama.
 - Če je indeks manjši od 0, se v seznamu nič ne spremeni.
 - Če je indeks večji ali enak velikosti seznama, se v seznamu nič ne spremeni.
 - Če je seznam prazen, se v seznamu nič ne spremeni.
- `vrni()`
 - Vrne podatek iz vozlišča.
 - Če je indeks manjši od 0, vrne nov/prazen objekt podatka.
 - Če je indeks večji ali enak velikosti seznama, vrne nov/prazen objekt podatka.
 - Če je seznam prazen, vrne nov/prazen objekt podatka.
 - Vračilo novega/praznega objekta podatka predstavlja napako.

Implementirajte

- `vrini()`
 - Vrine novo vozlišče v seznam.
 - Če je indeks manjši od 0, se v seznamu nič ne spremeni.
 - Če je indeks večji ali enak od velikosti seznama, vrine novo vozlišče na konec seznama.
 - Če je seznam prazen, vrine novo vozlišče na začetek seznama.
- `odstrani()`
 - Vrne podatek iz vozlišča in odstrani vozlišče iz seznama.
 - Če je indeks manjši od 0, vrne nov/prazen objekt podatka.
 - Če je indeks večji ali enak velikosti seznama, vrne nov/prazen objekt podatka.
 - Če je seznam prazen, vrne nov/prazen objekt podatka.
 - Vračilo novega/praznega objekta podatka predstavlja napako.

Testni scenariji

- vrini() in vstavi():
 - v neprazen seznam
 - v prazen seznam
 - če je indeks manjši od 0
 - če je indeks večji ali enak velikosti seznama
- vrni() in odstrani():
 - iz nepraznega seznama
 - iz praznega seznama
 - če je indeks manjši od 0
 - če je indeks večji ali enak velikosti seznama