

KURS JĘZYKA C++

KOLEJKA DWUSTRONNA

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

Zadanie.

Zdefiniuj klasę `lista` reprezentującą dwukierunkową listę z liczbami rzeczywistymi. Zakładamy, że elementy listy będą wirtualnie ponumerowane kolejnymi liczbami naturalnymi. Twoja lista powinna obsługiwać takie operacje jak wstawienie nowego elementu na zadaną pozycję do listy (pozycja 0 oznacza wstawienie na początek), usunięcie wskazanego elementu z listy, pobranie referencji do wskazanego elementu za pomocą operatora indeksowania oraz podanie rozmiaru listy.

Klasa `lista` ma być tylko opakowaniem dla homogenicznej struktury złożonej z węzłów (klasa `wezel`). Wszystkie operacje listowe wymienione wcześniej zaimplementuj w klasie `wezel` (możesz użyć rekurencji); Klasa `lista` ma za zadanie kontrolować początek i koniec listy i zlecać wykonanie operacji listowych skrajnym węzłom listy.

Następnie stosując niepubliczne dziedziczenie stwórz klasę reprezentującą kolejkę dwustronną `kolejka2stronna`, w której elementy można wstawiać i usuwać z obydwu stron. Struktura ta powinna implementować operacje wstawiania nowych elementów na początek i na koniec kolejki (oprócz funkcji składowych także za pomocą operatora dodawania `+`), usuwania elementu z początku i z końca kolejki (oprócz funkcji składowych także za pomocą operatora dekrementacji `--` w wersji prefiksowej i postfiksowej), podglądania elementu pierwszego i ostatniego oraz podawania liczby wszystkich elementów w kolejce.

W klasie węzła zdefiniuj operatory `new` i `delete`, które będą działały na pewnej globalnej tablicy (o określonym z góry rozmiarze), przeznaczonych tylko na obiekty typu `wezel`. Zdefiniuj też jakąś prostą klasę do zarządzania takim obszarem pamięci — należy pamiętać, które sloty w tablicy przeznaczonej na węzły są wolne a które zajęte (można wykorzystać do tego celu tablicę bitów z poprzedniego zadania).

Pamiętaj, aby w liście dwustronnej zdefiniować konstruktor domyślny, konstruktor kopiujący, konstruktor przenoszący, destruktor, przypisanie kopiujące, przypisanie przenoszące oraz zaprzyjaźniony operator pisania do strumienia (oraz ewentualnie czytania ze strumienia).

Na koniec napisz program, który rzetelnie przetestuje działanie Twojej kolejki.

Wskazówka.

Klasa `wezel` niech będzie klasą zagnieżdżoną w klasie `lista`. Klasa `wezel` ma posiadać w pełni zaprogramowaną funkcjonalność listy (w metodach klasy `lista` powinienś tylko wywoływać odpowiednie metody na obiekcie pierwszego/ostatniego węzła w liście).

Listę zaprogramuj w taki sposób, aby w stałym czasie umożliwić użytkownikowi dostęp do pierwszego i ostatniego elementu.

Uwaga.

Podziel program na pliki nagłówkowe i źródłowe.

Podpowiedź.

W funkcjach składowych i w konstruktorach zgłaszaj błędy za pomocą instrukcji `throw`.