

kurs języka C++

kolejka napisów

Instytut Informatyki
Uniwersytetu Wrocławskiego

Paweł Rzechonek

Prolog

Kolejka to bufor na dane, do którego można wkładać nowe elementy oraz wyciągać elementy, ale tylko w kolejności zgodnej z czasem ich włożenia – z kolejki można usunąć tylko ten element, który został do niej włożony najwcześniej. Odpowiednikiem tej struktury w życiu codziennym może być kolejka klientów do kasy w sklepie – klient, który ustawił się w kolejce jako pierwszy zostanie jako pierwszy obsłużony. Stąd się wzięła nazwa takiego bufora: FIFO (ang. *First In, First Out*), czyli „kto pierwszy przyszedł ten pierwszy wyjdzie”.

Zadanie

Zdefiniuj klasę `kolejka`, która będzie strukturą typu FIFO – element, który został do tej struktury dodany najwcześniej, będzie z niej wyciągnięty najszybciej. Struktura ta ma służyć do przechowywania napisów typu `string`.

Sama funkcjonalność kolejki ma być bardzo prosta:

- wkładamy napis do kolejki – funkcja składowa `void wstaw(string)`,
- wyciągamy napis z kolejki – funkcja składowa `string usun()`,
- podglądamy napis jest na początku kolejki – funkcja składowa `string zprzodu()`,
- pytamy o liczbę wszystkich elementów w kolejce – funkcja składowa `int dlugosc()`.

Kolejkę zaimplementuj na tablicy utworzonej dynamicznie na stercie w trakcie działania konstruktora (za pomocą operatora `new[]`, na przykład `kol = new string[pojemnosc]`). W destruktorze należy zwolnić pamięć przydzieloną dla tej tablicy (za pomocą operatora `delete[]`, na przykład `delete[] kol`). Implementując kolejkę na tablicy wykorzystaj zawinięcie tablicy (następny po ostatnim elemencie jest pierwszy element), co ułatwi obsługę tej struktury.

Pojemność kolejki ma zostać określona w konstruktorze – zdefiniuj więc prywatne pole `pojemnosc` typu `int`, w którym będzie pamiętany maksymalny rozmiar kolejki. Będziesz też potrzebować informacji o miejscu gdzie kolejka się zaczyna oraz o liczbie elementów aktualnie przechowywanych w kolejce – zdefiniuj zatem prywatne pola `poczatek` oraz `ile` typu `int`, w których będziesz pamiętać odpowiednio miejsce pierwszego elementu oraz liczbę elementów przechowywanych w kolejce.

```
class kolejka {
    int pojemność, początek = 0, ile = 0;
    string *tab;
    // ...
};
```

Kolejka ma posiadać pięć konstruktorów: konstruktor z zadaną pojemnością, konstruktor bezparametrowy i jednocześnie delegatowy (domyślna pojemność kolejki to 1), konstruktor inicjalizujący kolejkę za pomocą listy napisów (za pomocą `initializer list<string>`), konstruktor kopiujący i przenoszący. Aby uzupełnić semantykę kopiowania i przenoszenia zdefiniuj odpowiednie operatory przypisania (przypisanie kopiujące i przenoszące).

Na koniec napisz program rzetelnie testujący działanie kolejki (w szczególności przetestuj kopiowanie i przenoszenie).

Uwaga

Podziel program na pliki nagłówkowe (definicja klasy) i źródłowe (definicje metod zadeklarowanych w klasie). Funkcję `main()` z testami umieść w osobnym pliku źródłowym.

Ważne elementy programu

- Obiekt kolejki ma być inicjalizowany na kilka różnych sposobów: konkretną pojemnością, domyślnie (konstruktor delegatowy), przez skopiowanie z innej kolejki (konstruktor przenoszący), za pomocą listy wartości początkowych (lista wartości inicjalizujących zawartość kolejki), za pomocą przeniesienia zawartości z kolejki tymczasowej (konstruktor przenoszący).
- Obiekt kolejki ma być kopiowalny (przypisanie kopiujące i przenoszące).
- Zgłaszanie wyjątków w konstruktorach i funkcjach składowych.
- Przetestowanie klasy `kolejka` w funkcji `main()`.