

Programowanie i metody numeryczne, 26.03.2023

Krótki plan zajęć:

- Klasy, kontynuacja
- Smart pointer: `std::unique_ptr`
- Iteratory

Zadania:

Zadanie 1

Napisać funkcję `'concatenate'`, która jako argument przyjmie dwie tablice wraz z ich rozmiarami i zwróci nową tablicę będącą połączeniem dwóch pierwotnych tablic. W funkcji `main` zdefiniować dwie tablice na których zostanie przeprowadzona ta operacja, połączyć je w nową tablicę a następnie usunąć/zdealokować tablice pierwotne. Wykorzystać w tym zadaniu słowa kluczowe `new/delete`.

Kontynuacja: Zadanie 2

Napisać klasę wzorowaną na `std::vector` opartą o tablice. Klasa ta powinna mieć:

- konstruktor domyślny tworzący pustą tablicę jak i taką o zadanym rozmiarze oraz o zadanym rozmiarze wypełnioną podaną wartością
- zdefiniowany destruktork
- zdefiniowaną metodę `size()` zwracającą rozmiar wektora oraz metodę `print()` wypisującą zawartość danego wektora.
- zdefiniowaną metodę `push_back()`, która dodaje element na koniec wektora
- zdefiniowany operator `[]` zwracający referencję do n-tego elementu,
- zdefiniowany operator `«` pozwalający na przekierowanie wektora do strumienia wyjścia.

W pierwszej kolejności skorzystać z tworzenia tablicy opartego na metodach `new/delete`.

Zadanie 3

Zmodyfikować kod z zadań 1 i 2 korzystając z `unique_ptr`.

Zadanie 4

Napisać iterator dla uprzednio zdefiniowanej klasy. Iterator ten powinien zawierać:

- konstruktor przyjmujący wskaźnik jako argument
- zdefiniowany operator `*`
- zdefiniowany operator `->`
- zdefiniowany operator `++` oraz `++(int)`, innymi słowy lewo- i prawostronny operator `++`
- zdefiniowany operator `==` oraz `!=`

Tak zdefiniowany iterator można załączyć do własnej klasy wektor. Zdefiniować w klasie iterator `begin()` oraz `end()`.

Sprawdzić czy iterator spełnia swoje zadania i wypełnić wektor wartościami korzystając z `std::iota` albo zastosować `std::transform`.

Dopisać aliasy do iteratora: `'iterator_category'`, `'value_type'`, `'reference'`, `'pointer'`, `'difference_type'`.