

Zaawansowane C++

Lista 8: smart pointers

Zadanie 1

Napisz program zawierający definicje dwóch klas:

- `cpplab::fuel_tank` inicjowaną w konstruktorze ilością dostępnego paliwa (`unsigned int`). Klasa powinna mieć metodę `unsigned int refuel(unsigned int)`, która zwróci żadaną ilość paliwa (zmniejszając ilość w zbiorniku) jeśli jest dostępna lub zero w przeciwnym wypadku.
- `cpplab::engine` zawierającą listę podłączonych zbiorników z paliwem (lista odpowiednich smart pointerów) oraz metodę służącą do podłączenia zbiornika do silnika (dodającą smart pointer wskazujący na zbiornik do listy). Obiekt tej klasy w konstruktorze powinien startować wątek pobierający co zadany czas pewną ilość paliwa z jednego z dostępnych zbiorników (konstruktor ma mieć dwa argumenty - interwał i ilość paliwa). Jeśli ze zbiornika nie da się pobrać potrzebnej ilości paliwa, należy go odłączyć od silnika (usunąć z listy). Wątek powinien się zakończyć jeśli lista zbiorników przy próbie pobrania paliwa będzie pusta. Zbiorniki z paliwem mogą być współdzielone przez kilka silników.

W funkcji `main` stwórz trzy silniki pobierające:

- pięć jednostek paliwa co dwie sekundy,
- jedną jednostkę paliwa co sekundę,
- dwie jednostki paliwa co trzy sekundy,

oraz dziesięć zbiorników zawierających po nie mniej niż dziesięć jednostek paliwa, które zostaną podłączone do wszystkich silników i pozwolą na ich pracę przez co najmniej 10 sekund. W rozwiązaniu proszę nie stosować jawnie operatorów `new` i `delete`.

Zadanie 2

Napisz własną implementację `cpplab::unique_ptr` zawierającą podstawowe funkcjonalności unique pointera, które są opisane na `cppreference`.

Napisz wskaźnik `cpplab::non0_ptr`, który **nie** może przyjmować wartości `nullptr`.

Materiały pomocnicze:

- Short introduction from the Chernobyl
<https://www.youtube.com/watch?v=U0B7-B2MfwA&t=44s>
- Longer version from cppcon22
<https://www.youtube.com/watch?v=YokY6HzLkXs&t=191s>