



# POLSKO-JAPOŃSKA AKADEMIA TECHNIK KOMPUTEROWYCH

## PJC Zadania 11

Rozwiązania należy przesłać w postaci pojedynczego pliku o rozszerzeniu `.cpp`

### Zadanie 1

Napisz szablon funkcji `contains_all()`, przyjmującą dwa argumenty będące dowolnymi kontenerami. Funkcja winna zwrócić prawdę lub fałsz, zależnie od tego, czy w pierwszym przekazanym kontenerze znajdują się wszystkie elementy z drugiego przekazanego kontenera. Zwróć uwagę na krotność elementów.

### Zadanie 2

Stwórz przestrzeń nazw `pjc`, w której stworzysz szablon funkcji `swap()`. Jego celem będzie zamienienie wartościami podanych mu argumentów.

### Zadanie 3

Stwórz szablon klasy `container_wrapper` dziedziczący publicznie z `std::vector` przechowującego elementy typu parametryzującego klasę, którą właśnie tworzysz. Dodaj do niej omawiany na ćwiczeniach szablon metody `add_all()`, który przyjmie dowolny kontener i wszystkie elementy z niego doda do obiektu, na rzecz którego jest wywoływany. Następnie dodaj szablon `remove_all()`, który usunie z obiektu, na którym jest wywoływany, wszystkie elementy obecne w przekazanym mu argumencie.

### Zadanie 4 (dodatkowe)

Do szablonu `container_wrapper` dodaj bezparametrowy szablon metody `to()`, który precyzując szablon kontenera, jaki chcemy wyprodukować, stworzy go i zwróci. Wynikowy

obiekt winien przechowywać kopie wszystkich elementów, które obiekt, na rzecz którego został wywołany ten szablon, przechowuje.

```
auto ints = container_wrapper<int>();
ints.push_back(50);
ints.insert(ints.begin(), 5);
ints.add_all(std::vector<int>{20, 55, 45, 35, 30});
ints.push_back(25);
ints.insert(ints.begin(), 10);
ints.insert(ints.begin() + 2, 40);
ints.push_back(15);
ints.pop_back();
ints.push_back(15);

for (const auto& item : ints) {
    std::cout << item << ' ';
}

std::cout << '\n';
for (const auto& item : ints.to<std::set>()) {
    std::cout << item << ' ';
}
```

Wynik:

```
10 5 40 50 20 55 45 35 30 25 15
5 10 15 20 25 30 35 40 45 50 55
```

Warto zwrócić uwagę, że fakt posortowania elementów w drugiej linii wyniku wyłącznie z tego, że został stworzony `std::set<int>` na podstawie obiektu `ints` (zawierający wszystkie elementy z tego obiektu).

*Przykład 1: użycie szablonu `container_wrapper`, jego odziedziczonych metod oraz szablonu `to()`*