

# Języki i metody programowania 2

## Lab 6

### Przeciążanie operatorów

---

Przykładowe deklaracje przeciążenia operatorów (plik nagłówkowy):

```
//przeciążenie operatora dodawania
//pozwala na operacje studentYear + anotherStudentYear
StudentYear operator+(StudentYear &studentYear);

void operator++(int);

//tutaj wyjątkowo jest to funkcja zaprzyjaźniona by móc wywołać operator
//nawet gdy nie ma utworzonego obiektu
friend ostream &operator<<(ostream &output, const StudentYear & studentYear);

int operator[](int index)

//przeciążenie operatora rzutowania
//użycie w kodzie: static_cast<int>(studentYear);
operator int()
```

### Zadania

---

#### Zadanie 1

1. Napisz klasy:
  - a. Student - z polami id, firstName, lastName, field, year, gdzie wszystkie pola poza year są typu string, a pole year ma własny typ StudyYear
  - b. StudyYear - z polem value (typu int)
  - c. StudentRepository - będzie przechowywać kolekcję studentów
2. Zdefiniuj operator ++ i -- do inkrementacji/dekrementacji roku studiów
3. Zdefiniuj operator « i » zapisu i odczytu ze strumienia (jako funkcje zaprzyjaźnioną)
  - a. dla klasy StudyYear - zwraca rok studiów w postaci np. „year 1”
  - b. dla klasy Student - zwraca stringa w formie JSON (np. {id: „2030001234”, firstName: „Arkadiusz”, lastName: „Kowalski”, field: „informatyka”, year: 1}
4. Zdefiniuj operator == porównania i < mniejszości dla StudyYear
5. Zdefiniuj operator [] zakresu dla StudentRepository pozwalający na pobranie z repozytorium studenta o określonym id
6. Zdefiniuj operator rzutowania do typu int dla StudyYear.