

Laboratorium 5 Pomoc. Właściwości programowania obiektowego: hermetyzacja i dziedziczenie.

Cel laboratorium:

Zapoznani z podstawowymi właściwościami programowania obiektowego - hermetyzacją i dziedziczeniem.

- **Hermetyzacja** - ograniczenie dostępu do określonych pól/metod klasy w celu zabezpieczenia ich przed niepowołaną ingerencją z zewnątrz i błędnym użyciem
 - Realizacja: **private, protected, public**
- **Dziedziczenie** - tworzenie nowej klasy na podstawie jednej lub kilku utworzonych wcześniej klas bazowych
 - Rodzaje: jednokrotne/wielokrotne (nie jest zalecane)
 - Składniki dziedziczone: pola i metody z sekcji public i protected
 - Składniki nie dziedziczone: pola i metody z sekcji private, konstruktory, destruktory, przeciążany operator przypisania(=) wykorzystywany do kopiowania obiektów
 - Realizacja:

```
class NazwaNowejKl: [specyfikator] NazwaKlBazowej //jednokrotne
{
    //deklaracje składników nowej klasy
};
```

```
class NazwaNowejKl: [specyfikator] NazwaKlBazowej1, //wielokrotne
                  [specyfikator] NazwaKlBazowej2, ....
{
    //deklaracje składników nowej klasy
};
```

- Specyfikatory dostępu (rodzaje dziedziczenia):
 - **Public – dziedziczenie publiczne:** część public przodka jest public w potomku, część protected przodka jest protected w potomku - **relacja jest-czymś**
 - **Protected –** część public i protected przodka jest protected w potomku
 - **Private - dziedziczenie prywatne:** część public i protected przodka jest private w potomku - **relacja ma coś** (alternatywa zawierania)
- Dziedziczenie wielokrotne - nie można kilkakrotnie wykorzystywać danej klasy bazowej
- Przesłanianie metod - wywołanie metod przodka przez operator zakresu:

```
klasaprzodka::metoda (parametry)
```

- Konstruktory w typach potomnych

```
class A
{
    int pa;
    public:  A(int a);
};
A::A(int a)
{ pa=a; }
```

```
class B: public A
{
    int pb;
    public:  B(int a, int b);
};
B::B(int a, int b) :A(a)
{ pb=b; }
```