

## Metodyka i Techniki Programowania II

### C++: Dziedziczenie 2

Autorzy instrukcji: dr inż. Zbigniew Hulicki, mgr inż. Artur Kos

Kontynuacja tematu dziedziczenia w C++, w tym wielokrotne dziedziczenie, metody wirtualne.

### Zadanie 1

1. Rozszerz kod z poprzednich zajęć (zamieszczony poniżej):

```
#include <iostream>
using namespace std;

class student
{
private:
    string imie_nazwisko_ = "NO_NAME";
    unsigned int nr_indeksu_ = 0;
public:
    student(string imie_nazwisko, unsigned int nr_indeksu);
    string Opis_ = "student grupy";
    void printOpis();
    void printDane()
    {
        cout << " Metoda printDane klasy bazowej" << endl;
        cout << " imie nazwisko " << imie_nazwisko_ << endl;
        cout << " nr indeksu " << nr_indeksu_ << endl;
    }
};

class starosta : public student
{
private:
    string email_ = "nouser@noemail";
public:
    starosta(string imie_nazwisko, unsigned int nr_indeksu, string email);
    string Opis_ = "starosta grupy";
    void printDane()
    {
        cout << " Metoda printDane klasy starosta" << endl;
        cout << " email " << email_ << endl;
        student::printDane();
    }
};

starosta::starosta(string imie_nazwisko, unsigned int nr_indeksu, string email) :
    student(imie_nazwisko, nr_indeksu), email_(email)
{
    cout << "Tworzenie obiektu klasy starosta o nazwie: " << Opis_ << endl;
}

student::student(string imie_nazwisko, unsigned int nr_indeksu) :
    imie_nazwisko_(imie_nazwisko)
{
    nr_indeksu_ = nr_indeksu;
    cout << "Tworzenie obiektu klasy student o nazwie: " << Opis_ << endl;
}

void student::printOpis()
{
    cout << "Opis: " << Opis_ << endl;
}

int main()
{
    student stud("Jan Kowalski", 7);
    stud.printOpis();
    cout << "Dane:" << stud.imie_nazwisko_ << " " << stud.nr_indeksu_ << endl;
    starosta star("Aleksandra Nowak", 999, "mail@nomail.dot");
    star.printOpis();
    cout << "Dane:" << star.imie_nazwisko_ << " " << star.nr_indeksu_ << endl;
}
```

o dodatkową klasę bazową dla klasy **starosta** klasę **funkcyjny**. Klasa **funkcyjny** ma zawierać metody **poinformuj\_grupe** i **poinformuj\_prowadzacego**. Czy program się poprawnie kompiluje?

2. Utwórz nowy obiekt klasy **starosta** i uruchom program. Jaki jest rezultat ? Jaki z tego płynie wniosek? (jak działa konstruktor, jak działa metoda printDane())
3. W nowych metodach wywołaj wydruk stosownych komunikatów.
4. Następnie kod umieść **\*\*W SWOJEJ GAŁĘZI\*\*** w repozytorium zdalnym.

### **Dodatkowo**

**Dla początkujących:** Omówić działający kod z zadania 1. Wskazać konstruktor klasy, rodzaj konstruktora, metody klasy bazowej i pochodne, jak tworzone są obiekty klas.

**Dla zaawansowanych:** Spróbuj dopisać do klasy bazowej konstruktor kopiujący

### **Zadanie 2**

Korzystając z kodu z zadania 1:

1. Zmodyfikuj klasę student aby metoda **printOpis()** była metodą wirtualną.
2. Jak zmieni się działanie programu?
3. Następnie kod umieść **\*\*W SWOJEJ GAŁĘZI\*\*** w repozytorium zdalnym

### **Zadanie 3**

1. Uruchom poniższy kod:

```

#include <iostream>
using namespace std;

class urzadzenie
{
public:
    virtual int zapis(int id, string dane) = 0;
    virtual string odczyt(int id) = 0;
    virtual ~urzadzenie() {}
};

class dysk : public urzadzenie
{
private:
    int id_;
    string dane_;
public:
    dysk(int id);
    int zapis(int id, string dane);
    string odczyt(int id);
};

dysk::dysk(int id)
{
    cout << "Tworzenie obiektu klasy dysk " << endl;
    id_ = id;
}

int dysk::zapis(int id, string dane)
{
    dane_ = dane;
    cout << "zapis danych: " << dane << endl;
    return 0;
}

string dysk::odczyt(int id)
{
    cout << "odczyt danych: " << dane_ << endl;
    return dane_;
}

int main()
{
    urzadzenie u;
    dysk d1(7);
    d1.zapis(7, "test 11");
    d1.odczyt(7);
}

```

1. Czy program się kompiluje?
2. Popraw program i przeanalizuj jego działanie. Zwróć uwagę jak tworzony jest nowy obiekt klasy pochodnej.
3. Uzupełnij definicję klasy dysk aby następowało sprawdzanie identyfikatora **id** i w przypadku podania poprawnego identyfikatora wykonywane były operacje metody. W przypadku poprawnej pracy funkcja zapis ma zwracać wartość 1 w przeciwnym wypadku wartość -1.
4. Uruchom program.
5. Następnie kod umieść **\*\*W SWOJEJ GAŁĘZI\*\*** w repozytorium zdalnym

