

Zajęcia 6: Rozbudowa projektu – cz. III – funkcje operatorowe

1. Rozbudowa klasy **Data**

Dla klasy `Data` wykonaj następujące czynności:

- Opracuj funkcję operatorową (globalną – zaprzyjaźnioną z klasą `Data`) wstawiania do strumienia wyjściowego. Zadaniem tej funkcji jest wyprowadzenie informacji o danym obiekcie klasy `Data`:

```
ostream & operator<<(ostream & wy, const Data & d);
```
- Stwórz funkcję operatorową (globalną – zaprzyjaźnioną z klasą `Data`) pobierania ze strumienia wejściowego. Zadaniem tej funkcji jest, podobnie jak funkcji `Wpisz()`, wprowadzenie informacji do danych składowych obiektu klasy `Data`:

```
istream & operator>>(istream & we, Data & d);
```

2. Rozbudowa klasy **Napis**

W klasie `Napis` wykonaj następujące czynności:

- Zdefiniuj operator porównania w odniesieniu do klasy `Napis`.

```
bool operator==(const Napis & wzor) const;
```

 metoda ma zwracać `true`, gdy dane składowe obiektów są jednakowe, `false` w przeciwnym wypadku.
- Globalną, zaprzyjaźnioną z klasą `Napis` funkcję operatorową wstawiania do strumienia wyjściowego:

```
ostream & operator<<(ostream & wy, const Napis & p);
```

 Zadaniem tej funkcji jest wyprowadzenie informacji o danym napisie (zawartość pola `m_pszNapis`). Działanie funkcji przetestuj na strumieniach.
- Globalną, zaprzyjaźnioną z klasą `Napis` funkcję operatorową pobierania ze strumienia wejściowego:

```
istream & operator>>(istream & we, Napis & p);
```

 Zadaniem tej funkcji jest podobnie jak funkcji `Wpisz()` wprowadzenie informacji do danych składowych obiektu klasy. Działanie funkcji przetestuj na strumieniach.

3. Rozbudowa klasy **Pracownik**

W klasie `Pracownik` wykonaj następujące czynności:

- Zdefiniuj operator porównania w odniesieniu do klasy `Pracownik`.

```
bool operator==(const Pracownik & wzor) const;
```

 funkcja ma zwracać `true` gdy dane składowe obiektów są jednakowe, `false` w przeciwnym wypadku.
- Globalną, zaprzyjaźnioną z klasą `Pracownik` funkcję operatorową wstawiania do strumienia wyjściowego:

```
ostream & operator<<(ostream & wy, const Pracownik & p);
```

 Zadaniem tej funkcji jest wyprowadzenie informacji o danym pracowniku. Działanie funkcji przetestuj na strumieniach.
- Globalną, zaprzyjaźnioną z klasą `Pracownik` funkcję operatorową pobierania ze strumienia wejściowego:

```
istream & operator>>(istream & we, Pracownik & p);
```

 Zadaniem tej funkcji jest podobnie jak metody `Wpisz()` wprowadzenie informacji do danych składowych obiektu klasy. Działanie funkcji przetestuj na strumieniach.

4. Modyfikacja interfejsu

Po zdefiniowaniu w/w funkcji zmodyfikuj wcześniej opracowany interfejs. Dodaj możliwość zapisywania i odczytu tablicy pracowników z pliku.