

## Zadanie 9

---

Proszę przeczytać rozdziały 15 i 16 (na temat klas i zaprzyjaźniania funkcji oraz operacji wejścia/wyjścia) i napisać program którego schemat podany jest poniżej. Przyda się przejrzanie zadania siódmego.

Program tworzy listę osób, a następnie zapisuje ją na dysk w *formacie binarnym* (to znaczy: tworzy obiekt klasy `ofstream` i pisze za pomocą metody `write` tej klasy). Funkcja `writeList` pobiera wskaźnik do pierwszego węzła listy oraz nazwę pliku do utworzenia; tworzy (otwiera) ten plik, zapisuje dane wykorzystując metodę `writeOsoba` klasy `Osoba` i zamyka plik.

Następnie wywoływana jest funkcja `recreateList` pobierająca nazwę pliku: odczytuje ona dane z pliku, odtwarza listę i zwraca wskaźnik do jej pierwszego węzła.

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

class Osoba {
    char*  imie;
    int    rok;
    Osoba* next;

    // konstruktor domyslny i kopiujacy nie beda
    // uzywane; zostawiamy je niezaimplementowane
    // w sekcji prywatnej, ale piszemy, zeby zaznaczyc,
    // ze to swiadoma decyzja
    Osoba();
    Osoba(const Osoba&);
    // to samo dla operatora przypisania
    Osoba& operator=(const Osoba&);

public:

    friend void showList(const Osoba*);
    friend void writeList(const Osoba*,const char*);
    friend void deleteList(Osoba*&);
    friend Osoba* recreateList(const char*);

    Osoba(const char* imie, int rok, Osoba* next) {
        // ...
    }
}
```

```

void writeOsoba(ostream& s) const {
    // ...
}

~Osoba() {
    cout << "Del " << imie << "; ";
    // ...
}
};

void showList(const Osoba* head) {
    // ...
}

void deleteList(Osoba*& head) {
    // ...
    // funkcja zmienia wartosc head,
    // dlatego wskaznik jest przez referencje
    //
}

void writeList(const Osoba* head, const char* fil) {
    // ...
}

Osoba* recreateList(const char* fil) {
    // ...
}

int main(void) {
    cout << "Creating list:" << endl;
    Osoba* list = new Osoba("Dorota",1981,0);
    list =      new Osoba("Cecylia",1982,list);
    list =      new Osoba("Basia",1983,list);
    list =      new Osoba("Ania",1984,list);
    showList(list);

    // piszemy liste na dysk
    writeList(list, "z_09.dat");

    cout << "\nDeleting list:" << endl;
    deleteList(list);

    list = 0; // na wszelki wypadek!

    cout << "\nRecreating list:" << endl;
    list = recreateList("z_09.dat");
}

```

```
        showList(list);

        cout << "\nDeleting list:" << endl;
        deleteList(list);
    }
```

Powyższy program, po uzupełnieniu implementacji, powinien się skompilować i wypisać

```
Creating list:
Ania(1984) Basia(1983) Cecylia(1982) Dorota(1981)
```

```
Deleting list:
Del Ania; Del Basia; Del Cecylia; Del Dorota;
```

```
Recreating list:
Ania(1984) Basia(1983) Cecylia(1982) Dorota(1981)
```

```
Deleting list:
Del Ania; Del Basia; Del Cecylia; Del Dorota;
```

---

*Termin: do 26 stycznia (włącznie)*

---

Rozwiązania, w postaci **jednego** pliku źródłowego zawierającego treść programu, proszę wrzucać w systemie EDU do katalogu „Foldery zadań / Zadanie\_09”

Nazwą pliku powinno być nazwisko z dużej litery (bez polskich znaków); rozszerzeniem musi być `.cpp`, czyli np. `Malinowska.cpp`.