

## Laboratorium 3 – Destruktory

### Zad. 1

Stwórz klasę `Zabawka` o zmiennych `nazwa` (łańcuch znaków) i `nr` (liczba całkowita). Zdefiniuj konstruktor inicjalizujący podane zmienne następująco:

```
Zabawka(char* nazwa_0)
{
    nazwa = new char[20];
    strcpy(nazwa, nazwa_0);
}
```

Stwórz obiekt typu `Zabawka` za pomocą tego konstruktora.

### Zad. 2

W konstruktorze dopisz polecenie wypisujące na ekran komunikat o tym, że obiekt zwany `nazwa` został stworzony.

### Zad. 3

Stwórz zmienną globalną typu `int` `ile_zabawek`. W konstruktorze zwiększ wartość `ile_zabawek` o jeden, a następnie zainicjalizuj zmienną `nr` za pomocą tej zwiększonej wartości `ile_zabawek`. W komunikacie wypisywanym na ekran zawrzyj informację o tym, ile jest teraz wszystkich zabawek i jaki jest numer zabawki właśnie stworzonej.

### Zad. 4

Zdefiniuj **destruktor** `~Zabawka()`, w którym wartość `ile_zabawek` ma być zmniejszona o jeden. Dopisz wewnątrz definicji destruktora komunikat o tym, jaką `nazwę` nosiła niszczona zabawka, jaki był jej `numer` i ile zostało po tej operacji wszystkich zabawek.

### Zad. 5

Dopisz w destruktorze po komunikacie polecenie zwalniające zarezerwowaną pamięć na zmienną `nazwa`:

```
delete nazwa;
```

### Zad. 6

Zadeklaruj kilka obiektów typu `Zabawka` i otocz te deklaracje **klamrami**. Wewnątrz klamry, ale po deklaracjach dopisz polecenie wypisujące na ekran komunikat o liczbie istniejących aktualnie zabawek. Skopiuj ten sam komunikat za klamrę kończącą blok z deklaracjami. Porównaj wyniki.

### Pytania:

- Jaką nazwę musi mieć destruktor?
- Ile parametrów wejściowych może mieć destruktor?
- Czy destruktor można przeciążyć?