

Ćwiczenie 2

Tworzenie klas i obiektów

Cel ćwiczenia

Celem ćwiczenia jest praktyczne zapoznanie się z podstawowymi zagadnieniami dotyczącymi tworzenia klas i obiektów w języku C++.

Przygotowanie się do ćwiczenia

- Powtórzyć wiadomości z dwóch pierwszych wykładów (wprowadzenie do programowania obiektowego w C++, tworzenie klas i obiektów - w tym charakterystyka konstruktora i destruktor).
- Powtórzyć wiadomości praktyczne z pierwszych ćwiczeń (prosty przykład dotyczący tworzenia klasy `koło` jako programowego modelu figury geometrycznej).

Zadanie do wykonania

Proszę utworzyć klasę o nazwie `prostokat` będącą programowym modelem figury geometrycznej.

Klasa powinna zawierać następujące pola składowe:

1. pole o nazwie `szer` typu `float` przechowujące szerokość prostokąta.
2. pole o nazwie `wys` typu `float` przechowujące wysokość prostokąta.

Klasa powinna zawierać następujące funkcje składowe:

1. konstruktor bezargumentowy (tworzy obiekt i ustawia pola składowe na wartość 1).
2. konstruktor z dwoma argumentami typu `float` określającymi długości boków prostokąta (tworzy obiekt i ustawia pola składowe na wartości tych argumentów).
3. destruktor (w takiej postaci jak destruktor domyślny).
4. funkcje o nazwie `getSzer` i `getWys` (tzw. *getter*y zwracające wartości pól składowych, bezargumentowe).
5. funkcje o nazwie `setSzer` i `setWys` (tzw. *setter*y nadające określoną wartość polom składowym). Funkcje bez wyniku, powinny posiadać jeden argument w postaci nowej wartości pola składowego.
6. funkcja o nazwie `obliczPole` (bezargumentowa, oblicza i zwraca wartość pola powierzchni prostokąta).
7. funkcja o nazwie `obliczObwod` (bezargumentowa, oblicza i zwraca wartość obwodu prostokąta).

Uwaga: jeżeli w powyższych podpunktach jest mowa o tym, że "*funkcja zwraca...*", to znaczy, że chodzi o zwracanie wyniku działania funkcji przy użyciu instrukcji `return`.

Rozwiązanie zadania powinno zawierać pełny kod źródłowy programu, tzn.:

- deklarację klasy `prostokat`,
- definicję klasy `prostokat`,
- krótki program (funkcja `main`) zawierający:
 - utworzenie obiektu `p` klasy `prostokat` i uruchomienie dla takiego obiektu kolejno wszystkich funkcji składowych w celach testowych,
 - utworzenie obiektu klasy `prostokat` w sposób dynamiczny (za pomocą operatora `new`) i obliczenie pola powierzchni dla takiego obiektu.

Zadania dodatkowe

Zadanie 1

Proszę w obu konstruktorach oraz w destruktorze klasy `prostokat` umieścić polecenia wypisujące na ekranie następujące komunikaty tekstowe:

- w konstruktorze bezargumentowym polecenie:
`cout << "Utworzono obiekt konstruktorem bezargumentowym" << endl;`
- w konstruktorze z argumentami polecenie:
`cout << "Utworzono obiekt konstruktorem z argumentami" << endl;`
- w destruktorze polecenie:
`cout << "Zniszczono obiekt" << endl;`

Następnie proszę przetestować działanie poniższego programu i zaobserwować w jakich momentach działania programu obiekty są tworzone i niszczone (zasygnalizują to komunikaty tekstowe).

```
int main()
{
    prostokat p1;           // konstruktor bezargumentowy
    prostokat p2(2, 3);     // konstruktor z argumentami

    // dynamiczne tworzenie obiektu:
    prostokat *wsk = new prostokat(3, 4); // konstr. z argumentami

    {
        prostokat p3; // konstruktor bezargumentowy
    } // tutaj koniec zakresu widoczności obiektu p3

    p1.prostokat::~~prostokat(); // uruchomienie destruktora dla p1
    delete wsk; // dynam. usunięcie obiektu wskazywanego przez wsk
    return 0;
} // tutaj koniec zakresu widoczności obiektów p1, p2
```

Zadanie 2

Proszę utworzyć klasę o nazwie `rownanie` będącą programowym modelem równania kwadratowego postaci $ax^2 + bx + c = 0$.

Klasa powinna zawierać:

- pola składowe przechowujące wartości współczynników równania a , b , c .
- konstruktor z trzema argumentami określającymi wartości współczynników równania.
- funkcję składową wczytującą z klawiatury wartości współczynników równania.
- funkcję składową wypisującą na ekranie równanie w postaci matematycznej (gdzie zamiast symboli a , b , c pojawią się konkretne wartości współczynników).
- funkcję składową do rozwiązywania równania (rozpatrzyć sytuację, gdy wyróżnik równania jest większy lub równy zero - wtedy obliczyć rozwiązania oraz sytuację, gdy wyróżnik jest mniejszy od zera - wtedy tylko komunikat, że brak rozwiązań w dziedzinie liczb \mathbb{R}).

Zadanie 3

Proszę utworzyć klasę o nazwie `uklad` będącą programowym modelem układu dwóch równań z dwoma niewiadomymi postaci:

$$\begin{cases} ax + by = c \\ dx + ey = f \end{cases}$$

Klasa powinna zawierać:

- pola składowe przechowujące wartości współczynników układu a , b , c , d , e , f .
- konstruktor z sześcioma argumentami określającymi wartości współczynników układu równań.
- funkcję składową wczytującą z klawiatury wartości współczynników układu równań.
- funkcję składową wypisującą na ekranie układ równań w postaci matematycznej (gdzie zamiast symboli a , b , c , d , e , f pojawią się konkretne wartości współczynników).
- funkcję składową do rozwiązywania układu równań metodą wyznaczników Cramera (rozpatrzyć trzy warianty układu równań: oznaczony, nieoznaczony, sprzeczny).