

Programowanie obiektowe C++

Przeciążanie operatorów, konstruktory

Obiekty, konstruktory, operatory

1. Przykład

Dana jest klasa do opisu ułamków zwykłych i podstawowych operacji arytmetycznych na ułamkach:

```
class TUlamek
{
public:
    TUlamek();
    TUlamek(int licznik);
    TUlamek(int licznik, int mianownik);

    TUlamek operator + ();
    TUlamek operator - ();
    TUlamek operator = (TUlamek ulamek);
    TUlamek operator + (TUlamek ulamek);
    TUlamek operator - (TUlamek ulamek);
    TUlamek operator * (TUlamek ulamek);
    TUlamek operator / (TUlamek ulamek);

    string napisz();
    double wartosc();
    void uprosc();
    void ustawLicznik(int licznik);
    void ustawMianownik(int mianownik);

private:
    int licznik;
    int mianownik;

    void poprawZero();
    void poprawZnak();
};
```

w skład której wchodzi:

- prywatne pola: **licznik** i **mianownik** – do zapamiętywania wartości licznika i mianownika ułamka,
- bezparametrowy konstruktor domyślny **TUlamek()** pozwalający na zdefiniowanie wartości 1 w postaci ułamka zwykłego $\frac{1}{1}$,
- konstruktor pozwalający na zdefiniowanie liczby całkowitej **TUlamek(int licznik)** w postaci $\frac{\text{licznik}}{1}$,
- konstruktor pozwalający na zdefiniowanie ułamka zwykłego **TUlamek(int licznik, int mianownik)** w postaci $\frac{\text{licznik}}{\text{mianownik}}$,
- dwie metody **ustawLicznik()** i **ustawMianownik()** do zmiany wartości licznika i mianownika ułamka,
- metoda **napisz()** zwracająca wartość ułamka w łańcuchu znaków do wyświetlania,
- metoda **wartosc()** zwracająca numeryczną wartość ułamka,
- metoda **uprosc()** służąca do skracania ułamka (wykorzystuje algorytm szukania największego wspólnego dzielnika dwóch liczb),

- metoda **poprawZero()** służąca do zmiany zerowych ułamków na postać $\frac{0}{1}$,
- metoda **poprawZnak()** przenosząca znak minus z mianownika do licznika,
- operatory jedno argumentowe + i - do zmiany znaku ułamka,
- dwuargumentowe operatory arytmetyczne: +, -, * i /.

Wygeneruj nowy projekt C++ i skopiuj do pliku źródłowego projektu zawartość pliku tekstowego **ulamki.cpp** ze strony przedmiotu. Uruchom program, prześledź jego działanie.

2. Zadanie

Wykonaj kilka dowolnych operacji arytmetycznych na nowo zdefiniowanych obiektach typu **TUlamek**. Wykonaj dowolne operacje arytmetyczne na tablicach obiektów **TUlamek**.

Zadania dodatkowe

1. Opracuj typ obiektowy i program do przechowywania i podstawowych operacji arytmetycznych dla liczb zespolonych w postaci: $\mathcal{R} + \mathcal{I} \cdot i$.
2. Opracuj typ obiektowy i program do przechowywania i podstawowych operacji arytmetycznych dla długich liczb całkowitych (więcej niż 20 cyfr w liczbie).