

LAB 1

Celem zadania jest zdefiniowanie klasy **Ulamek** (jako prostej **struktury** z polami o zakresie **public**).

Klasa posiada

2 pola typu **int** reprezentujące **licznik** oraz **mianownik**

3 metody: **Wczytaj**, **Wypisz**, **Skracanie**

Dodatkowo należy przeciążyć **operator dodawania** ułamków (jako funkcja zewnętrzna)

Należy utworzyć projekt składający się z 3 plików:

ulamek.h, **ulamek.cpp**, **progmmain.cpp**

Plik **ulamek.h** powinien zawierać:

- ✓ **definicję klasy** (w postaci struktury z polami o zakresie public oraz 3 metody)
- ✓ **prototyp funkcji** realizującej **dodawanie** ułamków (funkcja zewnętrzna)
- ✓ **prototyp funkcji NWD** (funkcja zewnętrzna) zwracającej największy wspólny dzielnik 2 liczb całkowitych (będących jej argumentami) – funkcja ta powinna być wykorzystana w metodzie skracania ułamków

Plik **ulamek.cpp** powinien zawierać: implementację metod, funkcji i operatorów.

W pliku **progmmain.cpp** zademonstruj przykładowe możliwości swojej klasy **Ulamek**

Uwagi

- Metoda **Wczytaj** powinna wczytać pola ułamka, wymuszając by zaakceptowany mianownik był różny od 0;
po udanym wczytaniu pól wywołaj tutaj jeszcze metodę **Skracanie**
- Metoda **Wypisz** powinna wypisywać ułamek w postaci: **licznik / mianownik**
Jeśli licznik jest równy 0 lub mianownik jest równy 1, to wypisz tylko licznik
- Metoda **Skracanie** skraca ułamek oraz „usuwa” minus z mianownika

Wskazówka

Napisać funkcję pomocniczą NWD zwracającą największy wspólny dzielnik dwóch liczb.

Pseudokod

NWD(liczba całkowita a, liczba całkowita b)

Dopóki $b \neq 0$

$r :=$ reszta z dzielenia a przez b

$a := b$

$b := r$

zwróć a

- **Operację dodawania ułamków** zaimplementuj tak, by jako zwracany wynik otrzymać od razu ułamek skrócony, tj. po obliczeniu wyniku wywołaj jeszcze metodę **Skracanie**