

KURS PROGRAMOWANIA W C++

DŁUGIE LICZBY CAŁKOWITE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

Zadanie

Zdefiniuj klasę `LiczbaCalkowita` reprezentującą liczbę całkowitą o nieograniczonym zakresie. Liczby te powinny być kodowane binarnie w systemie *uzupełnienia dwójkowego U2*. Do pamiętania długich liczb całkowitych wykorzystaj klasę `TabBit` z poprzedniego zadania.

```
class LiczbaCalkowita
{
protected:
    TabBit *bity;
public:
    LiczbaCalkowita ();
    LiczbaCalkowita (int n);
    LiczbaCalkowita (const LiczbaCalk &lc);
    virtual ~LiczbaCalkowita ();
    LiczbaCalkowita & operator= (const LiczbaCalkowita &lc);
    // ...
};
```

Klasa `LiczbaCalkowita` powinna być wyposażona w konstruktor domyślny (wartość domyślna nowoutworzonej liczby ma wynosić 0), konstruktor inicjalizowany wartością typu `int`, konstruktor kopiujący, destruktor i operator przypisania kopiującego.

W klasie `LiczbaCalkowita` zdefiniuj operatory umożliwiające wykonywanie obliczeń arytmetycznych (dodawanie, odejmowanie, mnożenie, dzielenie, reszta z dzielenia i zmiana znaku). Trzeba przy tym pamiętać, że klasa reprezentująca długą liczbę może być dość duża i nie powinno się (o ile to nie jest konieczne) przekazywać jej przez wartość za pomocą stosu. Można więc zdefiniować po dwie wersje każdego operatora arytmetycznego: jeden jako funkcja zaprzyjaźniona a drugi jako składowy operator przypisania. Do obliczeń arytmetycznych mogą ci się przydać operatory bitowe oraz przesunięcia.

```
class LiczbaCalkowita
{
    // ...
public:
    LiczbaCalkowita operator- ();
    friend LiczbaCalkowita operator+ (const LiczbaCalkowita &ll, const LiczbaCalkowita &lp);
    LiczbaCalkowita & operator+= (const LiczbaCalkowita &lp);
    // ...
};
```

Zaprogramuj także zaprzyjaźnione operatory do czytania ze strumienia wejściowego `operator>>` i pisania do strumienia wyjściowego `operator<<` długich liczb całkowitych w postaci dziesiętnej.

```
class LiczbaCalkowita
{
    // ...
public:
    friend istream operator>> (istream &we, LiczbaCalkowita &lc);
    friend ostream operator<< (ostream &wy, const LiczbaCalkowita &lc);
};
```

Uwaga

Całą definicję klasy `LiczbaCalkowita` podziel na część nagłówkową i źródłową. Napisz też program testowy do sprawdzenia, czy zdefiniowana klasa działa poprawnie — oblicz i wypisz na standardowym wyjściu, ile to jest $100!$ i $\binom{200}{100}$, wyznacz wartość 128-ej liczby Fibonacciego i 256-ej liczby Catalana oraz sprawdź czy $\underbrace{111\dots1}_{32}$ jest liczbą pierwszą.