Рациональные числа

Ratio – отношение, дробь

$$r=rac{m}{n}; \qquad m\in\mathbb{Z}$$
 (целое), $n\in\mathbb{N}$ (натуральное)

Сложение и умножение

$$\frac{m_1}{n_1} + \frac{m_2}{n_2} = \frac{m_1 n_2 + m_2 n_1}{n_1 n_2}, \qquad \frac{m_1}{n_1} \times \frac{m_2}{n_2} = \frac{m_1 m_2}{n_1 n_2}.$$

Алгоритм Евклида нахождения Наибольшего Общего Делителя

ullet для двух ненулевых целых чисел a и b, цикл пока b
eq 0:

$$t \leftarrow b$$
$$b \leftarrow a \mod b$$
$$a \leftarrow t$$

• по завершению цикла а – искомый НОД

Задание І: запрограммировать рациональные числа как класс:

- Конструкторы:
 - \bullet инициализация числителем (m) и знаменателем (n)
 - инициализация целым числомконструктор по умолчанию
- Оператор вывода рационального числа на печать
- **3** Операторы унарные: и ++, -- (префиксные и постфиксные); и бинарные: +, -, *, /, +=, -=, *=, /=
- Функцию возведения в целую степень: pow()
- Опреобразования rational ← double:
 № конструктор-преобразование из double с заданной точностью
 - оператор преобразования рационального числа в double
- предусмотрите преобразование к несократимому виду
- для всех операций и функций должны быть тестовые примеры
- в пятом пункте возможно использовать explicit

Использование класса рациональных чисел

Вычисление чисел Бернулли: алгоритм Akiyama-Tanigawa

- Задаётся целое число п
- Цикл по *i* от 0 до *n* с шагом 1:

$$A[i] \leftarrow rac{1}{i+1}$$
 Цикл по j от i до 1 с шагом -1 : $A[j-1] \leftarrow j imes (A[j-1]-A[j])$

- ullet В конце получаем A[0], которое и есть B_n
- ullet На следующем слайде иллюстрация алгоритма и значения B_{1-16}

Задание II

- Написать функцию возвращающую vector<Rational> содержащий N первых чисел Бернулли
- Вычислить $B_{2k}, k = 1...8$

Треугольник Akiyama-Tanigawa

