

ДОМАШНЕЕ ЗАДАНИЕ №2 ПО ПРОГРАММИРОВАНИЮ НА C++

Задача 1 (2 балла). Напишите программу, которая удаляет из корректной программы на C++03 все комментарии (`/*...*/` и `//...`). Можно считать, что в программе не используются макросы. Исходный код программы на C++03 подается на стандартный поток ввода, результат работы должен быть отправлен в стандартный поток вывода.

Задача 2 (8 баллов). В пространстве имён `NPolynom` реализуйте шаблонный класс `TPolynom` («Многочлен от одной переменной») на основе контейнера `std::vector`. Тип коэффициентов многочлена передавайте в качестве параметра шаблона. Хранение коэффициентов должно быть плотным (то есть должны храниться все коэффициенты, в том числе и промежуточные нулевые). Предусматривать возможность работы с нецелыми или отрицательными степенями переменной не нужно. Сделайте следующее:

0. Напишите конструктор, создающий многочлен нулевой степени по указанному коэффициенту.
1. Перегрузите операторы `==` и `!=`. Ваш код должен быть очень простым.
2. Перегрузите операторы `+`, `-`, `*`, а также соответствующие операторы вида `@=`. Учтите, что должны быть определены и такие арифметические операции, в которых один из аргументов является скаляром.
3. Напишите функцию `Degree` для вычисления степени многочлена (считайте, что у нулевого многочлена степень равна -1).
4. Перегрузите операторы `[]` для получения (или изменения) коэффициента многочлена перед заданной степенью переменной. Оператор должен работать для любых степеней (в том числе больших текущей максимальной).
5. Перегрузите оператор `()` для вычисления значения многочлена в точке. Постарайтесь написать эффективный код.
6. Перегрузите оператор `<<` для печати многочлена в поток вывода.
7. Предусмотрите функции `begin` и `end` для доступа к итераторам (в т.ч. константного доступа), позволяющим перебрать или изменять коэффициенты многочлена. Учтите, что итератор должен иметь правильное состояние после того, как коэффициенты многочлена будут изменены.
8. Перегрузите операторы `/` и `%` для вычисления неполного частного и остатка от деления многочленов, а также соответствующие операторы вида `@=`. (считайте, что в этом случае деление коэффициентов допустимо). Учтите, что должны быть определены и такие арифметические операции, в которых один из аргументов является скаляром.
9. Перегрузите оператор `,` для вычисления наибольшего общего делителя (вызов будет делаться так: `(a, b)`).

Продемонстрируйте работу с объектами созданного класса на примерах. Реализация задачи обязана находиться в отдельном файле `polynom.h`. Подразумевается, что экземпляр класса будет создаваться так:

```
#include "polynom.h"
```

```
int main(int argc, char* argv[]) {  
    using namespace NPolynom;  
    TPolynom<...> a (...);  
    TPolynom<...> b (...);  
    TPolynom<...> c = (a, b);  
    c = a / b;  
    c *= 2;  
    ...  
}
```