



# Домашна работа 4

курс Обектно-ориентирано програмиране  
за специалност Информатика  
Летен семестър 2017/18 г.

## Полиноми

В последно време, особено под влиянието на ВА, полиномите станаха доста популярни. Те, разбира се, имат множество приложения - реално всяко компютърно изчисление малко или много се свежда до операции с полиноми... По този повод, но и от скука, екипът на курса по ООП реши да ви възложи следната задача:

Реализирайте клас полином, коефициентите на който да бъдат шаблонен тип. За класа предефинирайте поне следните операторни функции:

- Оператори за еквивалентност;
- Оператори за сравнение (един полином е по-голям от друг, когато е от по-висока степен);
- Бинарни оператори  $+$ ,  $-$ ,  $*$ ,  $/$  и  $\%$  с аргументи два полинома. Те извършват съответните действия, като деленето и остатъкът имат семантиката на целочислено деление и остатък в пръстена на полиномите. Реализирайте и съответните версии с присвояване;
- Бинарни оператори за умножение и деление с типа на коефициентите на полинома;
- Оператор за индексване, който връща коефициента пред съответната степен. Приемете, че всеки полином има безкраен брой коефициенти, краен брой от които са различни от 0;
- Оператор  $()$  с единствен аргумент  $X$  - от типа на коефициентите - пресмята стойността на полинома в точката  $X$ ;
- Оператор  $()$  с два аргумента  $a$  и  $b$  - от типа на коефициентите. Пресмята определен интеграл от полинома в граници  $[a, b]$ ;
- Оператори за инкрементиране, които пресмятат интеграл от полинома (приемете константата за 0);
- Оператори за декрементиране, които пресмятат производна на полинома;
- Оператори за вход и изход в поток;
- Оператор за преобразуване към цяло число - връща степента на полинома;
- Оператор за преобразуване към `bool` и прилежащ оператор `!` -> проверява дали полином е нулев.

Направете клас итератор, даващ възможност да се мине през всички не-нулеви едночлени на полинома, в нарастващ ред на степените им. За него предефинирайте операторните функции за:

- Инкрементиране и декрементиране;
- Сравнение (пълнен набор);
- Оператори за достъп до елемента (\* и ->).

Демонстрирайте използването на класовете в кратка програма.