**45.Програмни практики: припокриване на аритметични операции.**

Припокриващите се оператори са имплементирани като функции.Дефинират се като

‘operatorx’, където ‘x’ е операторът.

Например за припокриване с оператор за сумиране, можете да дефинирате функцията

като operator+.

Подобно е положението и при изваждане и равенство “-“ и “=”.

Операторите, използвани в с++ работят само с примитивни типове, не със наши

собствено създадени, тъй като компилатора незнае как да използва операторите за

нашите типове.

Всички традиционни оператори могат да бъдат припокривани, включително и new,

delete и конвертиращите оператори. Не е възможно да бъдат предефенирани ., .\*, ::, ?:

операторите.

Правила за припокриване:

- Не може да имаме нов оператор

- Не може да се променя броя на операндите, взимани от оператора.

- Не може да се променят приоритетите или асоциавността на операторите

За пример \* винаги има по-висок приоритет от +.

struct Complex {

....

Complex operator+( Complex &other );

};

Complex Complex::operator+( Complex &other )

{ return Complex( re + other.re, im + other.im ); }