

1. labor kis ZH, 2019. március 14., csütörtök 10-12

NÉV (olvasható):

NEPTUN kód:

Készíts komplex szám osztályt, amely a valós és a képzetes részt valós típusú tagváltozóban, az osztályon kívülről elérhetetlenül tárolja! Legyen alapértelmezett konstruktor, amely 0-ra állítja a számot, valamint olyan konstruktor is, amelynek megadható a valós és a képzetes rész értéke! (egyben is lehet). Készítsd el a következő három operátort:

+ operátor, mellyel két komplex szám összege számítható ki,

+ operátor, mellyel egy valós és egy komplex szám összege számítható ki (valós+komplex kell, a komplex+valós nem jó),

-= operátor, amely egy komplex számból kivon egy másik komplex számot.

Készíts main függvényt, melyben bemutatom a megírt függvények működését!

max. 10p, megoldási idő: 15 perc

```
#include <stdio.h>
#include <stdlib.h>

class komplex { // jó osztálydeklaráció: 1p
    double re, im; // privát tagváltozók: 1p
public: // publikus függvények: 1p
    komplex(double re = 0, double im = 0) : re(re), im(im) {} // két konstruktor: 1+1p
    komplex operator+(const komplex & b) const { return komplex(re + b.re, im + b.im); } // 2p
    komplex &operator-=(const komplex & b) { re -= b.re; im -= b.im; return *this; } // 2p
};

inline komplex operator+(double a, const komplex & b) { //2p
    return komplex(a) + b;
}

int main() { // 2p
    komplex a(1, 2), b(10, 20), c, d;
    c = a + b;
    d = 3.14 + a;
    b -= a;
    return 0;
}

// Összpont: ha szum(pontok) > 10, akkor 10, egyébként szum(pontok)
```