

NÉV (olvasható):

NEPTUN kód:

max. 10p, megoldási idő: 20 perc

1. Készíts generikus függvényt, amely két, azonos típusú elemekből álló tömböt kap paraméterként (különböző méretűek is lehetnek), és egy dinamikus tömbben visszaadja összefűzve a két tömböt, referencia paraméterben pedig az összefűzött tömb méretét! Írj kódrészletet, amelyben kipróbálsd a függvényt! Memóriaszivargás ne legyen! (6p)

```
template<typename T> // 1p
T * osszefuz(T * t1, size_t n1, T * t2, size_t n2, size_t & n_ossz) { // 1p
    n_ossz = n1 + n2;
    T * ossz = new T[n_ossz]; // 1p
    for (size_t i = 0; i < n1; i++) // algoritmus: 2p
        ossz[i] = t1[i];
    for (size_t i = 0; i < n2; i++)
        ossz[i + n1] = t2[i];
    return ossz;
}

double a[3] = { 1,2,3 }, b[4] = { 4,5,6,7 };
size_t uj_n;
double * t = osszefuz(a, 3, b, 4, uj_n); // 1p
for (size_t i = 0; i < uj_n; i++)
    std::cout << t[i] << ' ';
delete [] t; // -1p, ha nincs
```

2. Adott az alábbi Window osztály, amely egy téglalap alakú területet reprezentál a képernyőn. Származtass belőle gombot reprezentáló Button osztályt, amely pluszként a gomb feliratát tárolja! Készíts konstruktort, amely a gomb minden adatát inicializálja! Definiáld felül a print függvényt úgy, hogy az a gomb minden adatát írja ki! Írj kódrészletet, amelyben létrehozol egy gombot, és kiírod az adatait! A Window osztályt nem módosíthatod! (4p)

```
class Window {
    int x, y, w, h;
public:
    Window(int x, int y, int width, int height)
        : x{ x }, y{ y }, w{ width }, h{ height } {}
    virtual void print()const {
        std::cout << "x=" << x << ", y=" << y << ", w=" << w << ", h=" << h;
    }
    virtual ~Window() {}
};

class Button :public Window { // 1p
    std::string text;
public:
    Button(int x, int y, int width, int height, std::string text)
        : Window(x, y, width, height), text{ text } {} // 1p
    void print()const override {
        Window::print(); // 1p
        std::cout << ", text: " << text;
    }
};

Button g(1, 2, 3, 4, "OK"); // 1p
g.print();
```