

1. labor kis ZH, 2024. március 7., csütörtök 10-12

NÉV (olvasható):

NEPTUN kód:

- a) Készíts függvényt, amely megcseréli két, referencia paraméterben kapott valós változó értékét! (3p)
- b) Készíts függvényt, amely egy valós számokból álló tömböt vesz át, és egy új dinamikus tömbbe másolja a tömb tartalmát, fordított sorrendben, melyet visszaad! A dinamikus memóriakezelést a C++ operátoraival végezze! (4p)
- c) Írj main függvényt, amelyben kipróbáld az a) és b) feladatban létrehozott függvényeket! Memóriaszivargás ne legyen! (3p)

max. 10p, megoldási idő: 10 perc

```
#include <cstdio>

void csere(double& a, double& b) { // fejléc: 1p
    double c = a; // referencia helyes használata: 1p
    a = b; // függvénytörzs: 1p
    b = c;
}

double* fordit(double t[], int n) { // fejléc: 1p
    double* uj = new double[n]; // new 1p, használat 1p
    for (int i = 0; i < n; i++) // függvénytörzs: 1p
        uj[n - i - 1] = t[i];
    return uj;
}

int main() {
    double tomb[6] = { 1, 2, 3, 4, 5, 6 };
    csere(tomb[0], tomb[4]); // 1p
    double* t2 = fordit(tomb, 6); // 1p
    delete[] t2; // delete[] 1p, ha nincs []: 0p
    return 0;
}
```