

4. labor kis ZH, 2016. november 11., péntek 8-10

NÉV (olvasható):

NEPTUN kód:

Az alábbi két függvény közül elég az egyiket elkészítened. A másik függvénynek legalább a prototípusát meg kell adnod! A main függvényben viszont mindkét függvényt ki kell próbálnod, azt is, amelyiket nem csináltad meg. Ha mindkét függvényt elkészíted, akkor mindkettőre jár a pont, feltéve, hogy a rosszabbul sikerült függvényre kapható pontszám legalább felét eléred. Összesen maximum 10 pontot kaphatsz.

Írj egy függvényt, amely egy valós számokból álló tömbből kikeresi és visszaadja a legkisebb elem értékét, valamint az indexét is! (4+1p)

Írj függvényt, amely két karaktertömböt kap, és az elsőben lévő sztringet úgy írja a másodikba, hogy abban minden „s” betűt megdupláz! Azaz pl. be: „Esztergomban szépek a csillagok.”, ki: „Essztergomban sszépek a cssillagok.”. A nagy „S” betűket nem kell duplázni. (4+1p)

Írj main függvényt, amelyben kipróbálsz a két függvényt! (4p)

max. 10p, megoldási idő: 20 perc

```
#include <stdio.h>

double legkisebb(double *tomb, int n, int *index) {
    int i, min = 0;
    for (i = 1; i < n; i++) // tömb végigjárása 1p
        if (tomb[i] < tomb[min]) // minimum megtalálása 1p
            min = i;
    *index = min; // index visszaadása 0,5p
    return tomb[min]; // érték visszaadása 0,5p
} // tud két értéket visszaadni 1p

void duplaz(char *be, char *ki) {
    int i, j;
    for (i = j = 0; be[i] != '\0'; i++) { // bemenő sztring végigjárása lezáró 0-ig: 1p
        ki[j++] = be[i]; // külön index a két tömbnek 1p, eredeti átmásolása 0,5p
        if (be[i] == 's') // s duplázása 1p
            ki[j++] = 's';
    }
    ki[j] = '\0'; // helyes sztringkezelés 0,5p
}

int main() {
    double t[5] = { 3,2,1,4,6 }, min;
    char s[100];
    int index;
    min = legkisebb(t, 5, &index); // helyes függvényhasználat 1,5p
    printf("t[%d]=%g\n", index, min);
    duplaz("Esztergomban szepek a csillagok.", s); // helyes függvényhasználat 1,5p
    printf("%s\n", s);
    return 0;
} // egyebek: 1p
```