

4. labor kis ZH, 2014. november 14., péntek 8-10

NÉV (olvasható):

NEPTUN kód:

- a) Írj **függvényt**, amely kap egy sztringet, és kiszámítja a hosszát! (Nem használhatod a string.h függvényeit.)
- b) Írj **függvényt**, amely kap két karaktertömböt, és az első tömbben lévő sztringből a kisbetűket a második tömbbe másolja sztringként!
- c) Írj **kódrészletet**, amelyben definiálsz két karaktertömböt (s1,s2), az s1-et inicializárod az „Ez a sztring a bemenet.” szöveggel, majd átmásolod s1-ből az s2 tömbbe a kisbetűket, továbbá kiírod az s2-be másolt sztring hosszát és magát az s2 sztringet. Használd az a) és b) feladat függvényeit is! (Ha helyes a program, akkor s2-ben „zasztringabemenet” sztring lesz, melynek hossza 17 karakter.)

max. 10p, megoldási idő: 15 perc

```
int hossz(const char *s){
    int n;
    for (n = 0; s[n] != '\0'; n++)
        ;
    return n;
}

void kiskopi(const char *be, char ki[]){
    int i, j;
    for (i = j = 0; be[i] != '\0'; i++)
        if (be[i] >= 'a' && be[i] <= 'z')
            ki[j++] = be[i];
    ki[j] = '\0';
}

char s1[50] = "Ez a sztring a bemenet.", s2[50];
kiskopi(s1, s2);
printf("%d, %s\n", hossz(s2), s2);
```

4. labor kis ZH, 2014. november 19., Szerda 10-12

NÉV (olvasható):

NEPTUN kód:

- a) Írj **függvényt**, amely kap két karaktertömböt, és a másodikban lévő sztringet az elsőbe másolja! (Nem használható a string.h függvényeit.)
- b) Írj **függvényt**, amely kap két karaktertömböt, és egy logikai értéket ad vissza, amely IGAZ, ha a két sztring egymás fordítottja, és HAMIS, ha nem az. (Használható a string.h függvényeit.)
- c) Írj **kódrészletet**, amelyben definiálsz két karaktertömböt (s1,s2), s1-et inicializálsz egy szöveggel, majd a függvényével átmásolod s2-be, és kiírod, hogy a két sztring egymás fordítottja-e. A kiírt szövegnek tartalmaznia kell a sztringeket is! Használd az a) és b) feladat függvényeit is!

max. 10p, megoldási idő: 15 perc

```
void mycopy(char cel[], const char * forras){
    int i;
    for (i = 0; forras[i] != '\0'; i++)
        cel[i] = forras[i];
    cel[i] = '\0';
}

int forditotte(const char *s1, const char *s2){
    int hossz = strlen(s1);
    int i;
    if (hossz != strlen(s2))
        return 0;
    for (i = 0; i < hossz; i++)
        if (s1[i] != s2[hossz - 1 - i])
            return 0;
    return 1;
}

char s1[50] = "Abcdef fedcbA", s2[50];
mycopy(s2, s1);
if (forditotte(s1, s2))
    printf("%s forditottja %s\n", s1, s2);
else printf("%s nem forditottja %s\n", s1, s2);
```