Tartalomjegyzék

11. gyakorlat - C mutatók

Ezen a gyakorlaton az el?z? heti el?adás anyagát dolgozzuk fel.

Feladatok

Feladatok a CloudCoder-en megtalálhatóak. Ajánlott sorrend:

- kiegyenlit
- hanyszor
- sorhossz
- gyakorlatok
- kivalasztasos_rendezes

Feladat megoldások

kiegyenlit

```
void kiegyenlit(int *a, int *b) {
    if(*a >= *b + 2) {
        --*a;
        ++*b;
    } else if(*b >= *a + 2) {
        --*b;
        ++*a;
    }
}
```

hanyszor

A ciklusban mindig továbblépünk a szó el?z?leg megtalált helyér?l, miel?tt újra keressük a mondat hátralev? részében, hogy hol fordul el? legközelebb. Ezt addig tesszük amíg már nem fordul el? többször, akkor a *pos* értéke *NULL* lesz.

```
int hanyszor(const char *szo, const char *mondat) {
    const char* pos = strstr(mondat, szo);
    int db = 0;
    while(pos != NULL) {
        pos += 1;
        pos = strstr(pos, szo);
        ++db;
    }
    return db;
}
```

sorhossz

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

int main() {
    char sor[32];

    fgets(sor, 32, stdin);

    printf("%d", strlen(sor));

    return 0;
}
```

gyakorlatok

```
void kiegyenlit(int *a, int *b) {
    if(*a < *b - 1) {
        ++*a;
        --*b;
    } else if(*b < *a - 1) {
        --*a;
        ++*b;
    }
}

int main() {
    int n, letszam[20];
    int i, j;

    scanf("%d", &n);
    for(i = 0; i < n; ++i) {
        scanf("%d", &letszam[i]);
    }

    for(i = 0; i < 5; ++i) {
        int maxind = 0, minind = 0;
        for(j = 1; j < n; ++j) {
            if(letszam[j] > letszam[maxind]) {
                maxind = j;
            }
            if(letszam[j] < letszam[minind]) {
                minind = j;
```

```
            }
        }
        kiegyenlit(&letszam[minind], &letszam[maxind]);
    }

    for(i = 0; i < n; ++i) {
        printf("%d ", letszam[i]);
    }

    return 0;
}
```

kivalasztasos_rendezes

```
void csere(int *a, int *b) {
    int regi_a = *a;
    *a = *b;
    *b = regi_a;
}

int main() {

    int n;
    int tomb[20];
    int i, j;

    scanf("%d", &n);
    for(i = 0; i < n; ++i) {
        scanf("%d", &tomb[i]);
    }

    for(i = 0; i < n; ++i) {
        int minind = i;
        for(j = i; j < n; ++j) {
            if(tomb[j] < tomb[minind]) {
                minind = j;
            }
        }
        csere(&tomb[minind], &tomb[i]);
    }

    for(i = 0; i < n; ++i) {
        printf("%d ", tomb[i]);
    }

    return 0;
}
```

Házi feladatok megoldása

krampusz

```
void krampusz(int *a, int *b, int *c) {
    if(*a > *b + *c) {
        *a = 0;
    } else if(*b > *a + *c) {
        *b = 0;
    } else if(*c > *b + *a) {
        *c = 0;
    }
}
```

alfonz

```
int main() {
    int n, a, b, c, i;

    scanf("%d", &n);
    int sum = 0;
    for(i = 0; i < n; ++i) {
        scanf("%d%d%d", &a, &b, &c);
        krampusz(&a, &b, &c);
        sum += a;
    }

    printf("%d", sum);

    return 0;
}
```