Tartalomjegyzék

Elmélet

Google Drive

https://docs.google.com/document/d/1k8hpm1o9429d-Gs_OFAJMtkUWACGdYxvr0baZCnq6b8/edit

Ismétlés

- Dinamikus memória kezelés::

```
#include<stdlib.h>
...
int i;          // ciklusváltozónak
int m;          // ebbe olvassuk be a tömb méretét
scanf("%d",&m);
int *vec = (int *)malloc(m * sizeof(int)); // itt foglaljuk le a memóriát a tömbnek
...
for(i=0; i<M; i++){
  vec[i]=i*i;      // majd feltöltjük a tömböt az indexek négyzetével
}
...
```

File I/O, argumentumok

- El?adásjegyzet: File I/O
- fopen-el nyitunk meg file-t, fclose-al zárjuk be, FILE* segítségével dolgozunk rajta.
- Ugyanúgy írhatunk file-ba mintha printf-el tennénk csak fprintf-el kell és meg kell adni a file pointerét.
- Ugyanúgy olvashatunk file-ból mintha a terminálból, a felhasználótól olvasnánk be, csak fscanf-el és meg kell adni a file pointerét.
- Karakter tömbökbe olvashatunk a %s-el, ekkor az fscanf az elsõ whitespace karakterig olvas (space, újsor, tab...)
- Egy további példa:

```
#include<stdio.h>

int main(void){
```

```
  int i;
  int z;
  char s[100];
  FILE* fp;            // Letrehozzuk a file pointerunket
  fp = fopen("test.txt", "w");  // Megnyitjuk a test.txt-t irasra

  for(i = 0; i < 10; i++){
    fprintf(fp, "%d\n", i * i);  // Negyzetszamokat irunk a file-ba
  }
  fprintf(fp, "Most irunk a file-ba.\n");
  // Csak ugy irtunk valami szoveget a file-ba
  fclose(fp);    // Bezarjuk a file-t

  fp = fopen("test.txt", "r");    // Ujra megnyitjuk, de olvasasra

  for(i = 0; i < 10; i++){
    fscanf(fp, "%d", &z);         // Kiolvasunk a file-bol egy int-et
    printf("%d, ", z);            // Kiirjuk a kepernyore amit kiolvastunk
  }
  fscanf(fp, "%s", s);  // Kiolvasunk egy szot a file-bol
  printf("%s", s);      // Majd ezt ki is irjuk
  fclose(fp);     // Bezarjuk a file-t

  return 0;
}
```

- Argumentumokról: el?adásjegyzet

Feladatok

Mátrixok, 2 dimenziós tömbök

- Írj függvényt, mely dinamikusan lefoglal egy mátrixnak megfelel? helyet a memóriában.
- Írj függvényt, mely kiírja egy 2 dimenziós, dinamikusan lefoglalt mátrix elemeit a képerny?re.
- Dinamikus memóriafoglalással számold ki egy mátrix transzponáltját és négyzetét is.
- Írj függvényt, mely összead / kivon egymásból / összeszoroz 2 mátrixot.
- Megoldásaidat példákkal szemléltesd.

File I/O

Mátrixok

- Írj függvényt, mely file-ból beolvas a fenti függvényekkel használható mátrixot.
- Írj függvényt, mely file-ba ír egy megadott mártixot, használd az el?z? feladat kódjait.

Gyakoriság, hisztogram

- Írj függvényt, mely megszámolja egy egész számokból álló dinamikus tömb elemeinek a gyakoriságát.
- A függvény paraméterként kapja meg a vizsgált tömböt. Feltételezzük, hogy a tömbben N=20-nál kisebb, vagy egyenl? értékek szerepelnek (miért van erre szükség?)
- Oldjuk meg a feladatot tetsz?leges N-re.
- Írj függvényt, mely az adatokat file-ból olvassa be.