

Programozói dokumentáció: Határidőnapló

Adatszerkezetnek listát választottam, mivel ez a leghatékonyabb a feladat megoldásához, könnyű hozzáadni, elvenni elemeket belőle, illetve keresni is. Dinamikusan lehet területet foglalni vele, és nem kell előre megadni egy felső határt az elemszámnak. Benne elég rendezetlenül tárolni az elemeket, mivel a feladat egyik része sem indokolja, hogy egy adott sorrendben legyenek. Egy rekordhoz tartozó adatokat így láncolt listában tárolja, amiben négy szám adat, és három sztring van. A lista nem strázsás, egy elejére mutató pointer van, ami kezdetben NULL, így egy üres listát hozva létre. A listaelemek nincsenek rendezve, beolvasás szerint következnek egymás után. Egy ilyen struktúra:

```
typedef struct lista{
    int honap;
    int nap;
    int ora;
    int perc;
    char hely[51];
    char elnevezes[51];
    char megjegyzes[101];
    struct lista *kov;
}lista;
```

Main.c: Beolvasás, mentés, főmenü megjelenítése függvények:

- void fomenuoptions()

Kiírja a főmenüben választható lehetőségeket.

- void listabeolvas(lista **eleje)

A program elején (a main először ezt hívja meg) lehetőség van beolvasni egy szövegfájlból adatokat, a fájl nevének beírásával. (Ha abban a megfelelő formátum szerint szerepelnek az adatok:

“honap.nap.;ora;perc;hely;elnevezes;mjegyzes;”.) Ezeket a lista végére fűzi. Paraméterként megkapja a lista elejére mutató pointert, ha üres lenne a lista, meg is változtatja.

- void ment(lista *eleje)

Beolvassa egy szövegfájl nevét, majd abba (a megfelelő formátumban a későbbi kiolvasáshoz) belementi az adatokat. Mindegyiket külön sorba, ahogy végigmegy a listán, aminek az első elemére mutató pointert megkapja paraméterként, azt és a listát, nem módosítja.

- lista *beolvas()

Beolvas egy listaelemet, ha megfelelően van beírva. Memóriát foglal neki, belementi az adatokat, majd visszatérésként átadja az új listaelemet.

- int main()

Létrehoz egy üres listát, majd meghívja a listabeolvas függvényt.

Ezután a "főmenüben" marad végig, meghívja a fomenuoptions függvényt, majd mindig beolvassa a megfelelő menüpont sorszámát a 'szam' változóba. E szerint lép tovább és hívja meg a megfelelő függvényeket, vagy kilép a programból. A kilépésnél a felszabadítja egyesével a lista elemeit. Az egyes főmenüpontban pedig a beolvas függvény meghívása után, azt hozzáfűzi a lista végére.

Kereses modul: Kereső függvények:

- void kiirtalalat(int *szamoz, lista *mozgo)

Növeli a kiírt adat sorszámát, majd kiírja a 'mozgo' által mutatott listaelem adatait sorszámozva.

- void valasztas(int *mit, int *ezt)

Kiválasztja a felhasználóval melyik sorszámú elemet ('ezt'-ben tárolja), illetve annak melyik adatát módosítja('mit').

- void idokeres(lista *eleje)

A paraméterként megkapott listával dolgozik, abból keres hónap, nap vagy hét szerint. Ezek közül a sorszám beírásával lehet választani, ami a 'sorszam' változóban lesz eltárolva. Utána beolvassa mi alapján keressen keres, majd meghívja a kiirtalalt függvényt minden megtaláltnál.

Majd a visszalepbe beolvassa a '-1'-et visszatér a main-be.

Hét alapján való keresés esetén, megszorozza a keresett hetet 7-tel (napokban tárolja ezután), majd addig adja hozzá a hónapok napjait (honapszam tömbben tárolja a az egyes hónapok napszámait), amíg az több nem lesz, mint a 'napok', ezt a az 'osszeg'-ben tárolja.. Utána az utolsót

levonja az 'összeg'-ből, mivel az már több lenne. Hogy hányszor adta hozzá a hónapok napjait ('i' változó), mutatja melyik hónapban van a keresett hét. A szorzatból ('napok') kivonva a napok összegét('összeg'), pedig megkapja az utolsó napot a keresett hétből, ezt a 'konkretnap' tárolja.

- void nevkeres(lista *eleje)

Ugyan úgy keres, mint az idokeres, csak itt egy sztringet olvas be a 'kereses' karaktertömbbe, és az alapján keres. Kiirtalalat függvényt használva a kiíráshoz.

- lista *keres(lista *eleje,int menupont)

Kiírja a keresési lehetőségeket sorszámozva, amit majd a 'sorszam'-ban tárol. E szerint lép tovább és olvas be számokat('ho'/'o' és 'n'/'p')/sztringet('keres'), majd keres elemenként kiírva sorszámozva a megfelelőeket (meghívva a kiirtalalat függvényt) a paraméterként megkapott listából. Ezután attól függően, hogy a mainben melyik sorszámmal let meghívva ('menupont') hívja meg a menu2, illetve a torol függvényeket.

Cserel modul: Módosítás függvények:

- void szamvaltoztat(int *melyik,int ezt,lista *mozgo,int elso, int masodik, int keres)

Ha a 'melyik' eléri az 'ezt' sorszamú adatot megváltoztatja a 'mozgo' által mutatott listaelem számadatait, arra, amiket paraméterként kapott('elso', 'masodik').

- void sztringvaltoztat(int *melyik,int ezt,lista *mozgo, char *elso, int keres)

Ugyan azt csinálja, mint a szamvaltoztat, csak itt sztringet változtat.

- void menu2(int szam,int ezt,int sorszam,lista *eleje,int egyes,int kettes,char *keres)

Paraméterként megkapja a 'szam' változóba milyen adatát változtatja a kiválasztott elemnek(aminek a sorszámát az 'ezt'-ben kapja meg).

Ez alapján kéri, hogy írja be a felhasználó a megfelelő új adatokat (számokat: 'elso', 'masodik' és sztringet: 'elso'-ben tárolja). Majd a 'sorszam'-ban megkapott keresés sorszáma szerint, és a megkapott keresési

adatok (számok: 'egyes', 'kettes' vagy sztring: 'keres') alapján keresi a megfelelőeket, majd, ha annak sorszáma ('melyik') megegyezik az 'ezt'-tel módosítja a megfelelő adatot a sztringváltoztat vagy a számváltoztat meghívásával.

Torol modul: Törlés függvények:

- lista *eltavolit(lista *eleje,int *melyik, int ezt, lista *lemarado, lista *mozgo)

Ha a 'melyik' eléri az 'ezt'-ben tárolt sorszámot, akkor free-eli az adott listaelemet, amire éppen a 'mozgo' mutat. És átmutatja a pointereket a megfelelőre.

- void torol(int sorszam,lista **eleje,int egyes,int kettes,char *keres)

Paraméterként megkapja mi alapján keresett a felhasználó a keres függvényben a 'sorszam'-ban, és a keresési adatokat az('egyes', 'kettes' és a 'keres'-ben). A kilistázott rekordok közül választ a felhasználó, majd annak sorszámát az 'ezt'-ben tárolja.

Majd a sorszám alapján kikeresi a megfelelő sorszámút, a 'melyik'-ben számolja a megtaláltakat. Meghívva a eltavolit függvényt törli a megfelelőt.