

Határidőnapló- programozói dokumentáció

A program célja

Konzolban jelenít meg egy egyszerűen kezelhető határidőnaplót, angol nyelven.

A program fájljai

menues – A program menüit tartalmazza.

functions – A program fő függvényeit tartalmazza, **helperfunctions**

– A program segédfunkcióit tartalmazza.

A program struktúrái

A program három fő struktúrából áll, melyek egymásba is ágyazódnak.

A **Datum** tartalmaz három integert pontosabban évet, hónapot és napot melyek felhasználásra kerülnek az Esemeny és ListaElem struktúrában is, majd a datumCMP függvény is felhasználja ezt.

Az **Esemeny** struktúra tartalmaz egy dátumot (A Datum struktúra alapján) és két karaktertömböt, mely megadja a címet és a leírást. Ezek mérete nem adott a szövegbeolvas függvény tölti fel őket egyesével majd.

A **ListaElem** struktúra egy láncolt listát határoz meg, mely tartalmaz egy eseményt és egy következő eseményre mutató pointert.

```
typedef struct Datum {  
    int ev, honap, nap;  
}  
Datum;  
  
typedef struct Esemeny {  
    Datum d;  
    char *cim;  
    char *leiras;  
}  
Esemeny;
```

A program menük

ListaElem *menu(ListaElem *eleje, bool* vege) -Az alap menü, ami megjelenik amikor konzolban elindul a program. Itt hat lehetőség közül választhatunk, mely switch case-el és egy if résszel működik. Ha nem megfelelő sorszámot választunk azt is jelzi. Minden case függvényeket hív, vagy egy következő menüt. Átadódik neki a lista eleje, hiszen az általa hívott függvények mind azzal dolgoznak. A bool vege azért kell, hogy megmondja, hogy amikor mikor választjuk a hatos menüpontot, akkor mentsen a program a fájlba és álljon le.

void searchEventMenu(ListaElem *eleje) – A menü után töltődik be, ha a keresés opciót választjuk. Keresni kétféle képpen lehet, dátum vagy cím szerint. Választás után a menü a következő két függvényt hívhatja: searchbydate vagy searchbytitle. Ez a kiválasztás annyira egyszerű, hogy if-else ággal működik. Megkapja a listaelem elejét, mivel az általa hívott függvények teljes listával dolgoznak.

void listEventsMenu(ListaElem *eleje) – Az alap menüből töltődik be, ha a listázást választjuk. Itt négy lehetőségünk van kilistázni az eseményeket: év (listbyyear), hónap (listbymonth), nap (listbyday) és intervallum (listbyinterval) alapján. A kiválasztás switch case alapján történik, akkor is visszajelez, ha nem jó sorszámot választunk. Mindig egy következő függvényt hív.

Függvények

A főmenü függvényei

ListElem *scanEvent(ListaElem *eleje) – Új eseményt olvas be először bekérve a dátumot, a címet majd a megjegyzést. Utolsó kettőt dinamikusan foglalja a szovegbeolvas függvény segítségével. Ezután pedig az esemény a láncolt listába kerül, dátum szerint rendezve a datumCMP segítségével. Visszatér a lista elejével. Akármilyen dátumot elfogad, még a nemlétező hónapokat is.

ListElem *deleteEvent(ListaElem *eleje) – Először bekéri a törölni kívánt esemény dátumát és címét (a címhez a szovegbeolvas függvényt használva), ha pontos az egyezés csak akkor töröl. Az egyezést egy mozgó pointer segítségével keresi a listában. Ezen kívül tartozik hozzá egy lemaradó pointer is, hogy ne veszítsük el a listát. Képes a lista elejéről, közepéről és végéről is törölni. Ha nem találja a törlésre szánt elemet, akkor jelez. Visszatér a lista elejével.

ListElem *editEvent(ListaElem *eleje) – Csak teljes eseményt módosít, annak részeit nem. A deletevent segítségével megkeresi és törli a változtatni akart eseményt és a scaneventtel újat olvas be helyette.

A searchevent függvényei

void searchByDate(ListaElem *eleje) – Bekéri a pontos dátumot, amit szeretnénk keresni. Először az évet, majd hónapot és a napot. Egy mozgó pointer segítségével keres. Ezután kiírja a megtalált listaelemeket. Esemény változtatásra nincs lehetőség. Ha nem találja visszalép a menübe.

void searchByTitle(ListaElem *eleje) – Bekéri a pontos címet, mely alapján keresünk. Majd kiír minden olyan listaelemet, mely ezzel a címmel rendelkezik, ezeket úgy találja meg, hogy végigmegy a listán. Ha nem találja visszalép a menübe. Eseményt változtatni itt szintén nem lehet.

A kiírás mindkét esetben így történik:

```
printf("%d.%d.%d %s - %s\n", mozgo->esemeny.d.ev, mozgo->esemeny.d.honap, mozgo->esemeny.d.nap,
      mozgo->esemeny.cim, mozgo->esemeny.leiras);
```

A listeventsmenu függvényei

void listByYear(ListaElem *eleje) – Bekéri azt az évet, amit listázni akarunk. Egy mozgó pointer segítségével megkeresi a listaelemeket és kiírja azokat. Ha nem talál egyezést visszalép a menübe.

void listByMonth(ListaElem *eleje)– Bekéri előbb az évet, utána a hónapot, amit listázni akarunk. Szintén mozgó pointerrel végigmegy a listán. Ha talál megfelelő eseményt azokat kiírja. Ha nem akkor visszalép a menübe.

void listByDay(ListaElem *eleje)– Bekéri előbb az évet, utána a hónapot végül a napot, amit listázni akarunk. Ha talál megfelelő eseményt azokat kiírja. Ha nem akkor szintén visszalép a menübe.

void listByInterval(ListaElem *eleje)– Két dátumot kér, mindkettőnél először évet, hónapot majd napot. Listáz minden olyan eseményt, ami a két dátum közt van. Ezeket a datumCMP függvény és egy mozgó pointer segítségével teszi.

A fenti függvényeknél egyiknél sincs lehetőség az esemény változtatására. A kiírás úgy történik, mint más kiíró függvénynél.

Segédfüggvények

ListElem *readFromFile() - A függvényt a main hívja meg a program indulásakor, ez megnyitja a fájlt append+-al. A fájlból beolvassa listaelemeknek az eseményeket. Ha a fájl üres, az eleje pointer NULL lesz, ha nem, akkor egyenként feltölti. A fájlban minden új sorokkal van elválasztva. A dátumhoz fscanf-et használ, a címhez és a leíráshoz a filebolStringOlvas függvényt hívja segítségül. Kilépés előtt bezárja a fájlt, majd visszatér a listaelem elejével.

char *filebolStringOlvas(FILE *fp) -Segédfüggvény a cím és leírás fájlból való olvasásához. Egyenként foglal helyet a memóriában minden betűnek és visszatér a beolvasott szöveggel.

char *szovegbeolvas() – Hasonlóan működik, mint az előző függvény, ez viszont a scanEvent, a searchByTitle és a deleteEvent segédfüggvénye. Visszatér a beleírt szöveggel.

bool datumCMP(Datum a, Datum b) – Két dátumot hasonlít össze. Először az évet, majd a hónapot, végül a napot. Hamisat ad, ha az első dátum kisebb, mint a második.

void saveToFile(ListaElem *eleje) – Megnyitja a fájlt írásra és belemásolja a listaelemeket. A fájlt teljesen felülírja. Az írás minden elemet új sorba tesz. A main hívja ha a hatos pontot választjuk a menüben (tehát ha vége=false).

void felszabadit(ListaElem *eleje) – A teljes listát felszabadítja, a címet és a leírást is. A main hívja ugyanabban az esetben, mint amikor a saveToFile függvényt.