

# Határidőnapló dokumentáció

A programot Makefile segítségével fordítjuk. A fordításhoz elengedhetetlen egy C fordító. GCC 13.3.1 használatával lefordul a program hiba nélkül. A “make” parancs egyszeri lefuttatása után megjelenik a lefordult program “main” néven. Parancssorban a “./main” karakterszorozat lefuttatásával el tudjuk indítani a frissen generált programot. Módosítások után érdemes újrafordítani a forráskódot, hogy frissüljön a futtatható program.

## Precompile hasznosságok

- DEBUG  
Értéke 0 vagy 1 (igaz-hamis). Fel lehet használni olyan funkciókhöz, amiket nem feltétlen szeretnénk a programba belefordítani. Példa a main() függvényben van.
- MENU\_DEFINITION  
Makró. A menu/menu.h fájlban érdemes ezzel menüket definiálni.
- MENU  
Makró. C forrásfájlokban érdemes ezzel menüket definiálni.
- CLEAR\_SCREEN  
Makró. ASCII escape karakterszorozattal törli a parancssor/terminál kimenetét.
- WAIT\_FOR\_ENTER  
Makró. Menüben használatos, Enter gombnyomásra vár a menüciklusban.

## Struktúrák, konstansok

- Date  
Évek, hónapok, napok számát tároló struktúra.
- Time  
Órákat, perceket tároló struktúra.
- Event  
Az adatbázisban tárolt/tárolandó eseményeket reprezentálja ez a struktúra. Tárol dátumot, időt, címet, helyet, leírást.
- EventListNode  
Esemény láncolt lista elem. Eseményt és a következő elemre mutató pointert tárolja.
- State  
Tartalmazza az egész program állapotát. Tárolja az adatbázisfájl nevét és a láncolt listát, ami az eseményeket tárolja.

## Függvények, eljárások

Menük nem kerülnek ide, ugyanúgy működik az összes.

- int main(int argc, char \*\*argv)  
A program kezdőpontja. Beolvassa a parancssorból a paramétereket, flageket és aszerint indítja a programot. Inicializálja az állapotot és megkezdi a menüvezérlést.
- void print\_help()  
Kiírja a help menüt. A “--help” flaggel lehet program indításakor ezt előhozni.
- int parse\_date(Date \*d, char \*str)  
Dátumot állít elő egy szövegből.
- int parse\_time(Time\* t, char \*str)  
Időt állít elő egy szövegből.
- int parse\_event(Event \*e, char \*str)  
Eseményt állít elő egy szövegből.

- void free\_event\_strings(Event \*e)  
Egy esemény dinamikusan allokált szövegmezőit szabadítja fel.
- void free\_event\_list(EventListNode \*head)  
Egy dinamikusan allokált eseményláncotlistát szabadít fel.
- int restore\_state(State \*state, char \*filename)  
A program állapotát állítja vissza egy fájlból fájlnév alapján.