

Dokumentáció

A sakk programom félkész állapotánál a grafikus megjelenítés és a bábuk léptetése működik, és sárga négyzettel jelzi a lépési helyeket egy adott bábúnál.

Eddig működik a megjelenítés és kattintás, de azon belül a megjelenítésnél az `alap_hatter` függvény és a `hatter_rajzol` függvény érdekes bugot alkot vagy működést végez. Az `alap_hatter` egy 8x8-as bool tömb minden indexénél true-ra állít. Majd a `hatter_rajzol` függvény akkor rajzol ki egy adott sárga négyzetet, ha true. Ennek ellenére true-ra állított tömb elemeket nem jelzi ki sárgaként és a false-ra állított tömb elemeket jelzi ki sárgaként. Ezen a funkció működését még később kijavítom, de egyelőre működik fura viselkedése ellenére.

De nem csak ez a kettő függvénnyel és programrésszel gyarapodott a program. Felépítésében eddig a `main.c`, a `fv.c` és `fv.h` illetve a `lepes.c` és `lepes.h` fájlokból áll. A `main.c`-ben fut a fő programrész és itt főleg a grafikus megjelenítés logikája található jelenleg. A `fv.c`-ben grafikus és különböző általánosan kellendő függvények helyezkednek el, amit a `fv.h`-val csatolok össze a `main.c`-hez. A `lepes.c` fájlban a bábuk lépéseit leíró függvények vannak és a `lepes.h` függvénnyel fűződik a `main.c`-hez. Ezen fájlok mellé tervezek egy `logika.c` fájlt is létrehozni, amiben a sakk logika elemeit tervezem hozzáadni, illetve egy fájlkezelő fájlt is, ami a játék mentésében segít majd.

A `main.c` fájlban először a `kep_betolt` függvénnyel 3 képet töltök be az `img` mappából. Ezek után létrehozom a 8x8-as sakk_tabla, ami a bábuk helyzetét írja le. Létrehoz egy bool tömböt is, ami a sárga négyzetek megjelenését fogja szabályozni. Majd mindkét tömböt alaphelyzetre állítunk az `alap_hatter` és `alap_tabla` függvényekkel. A többi függvény a beállítja a megjelenő ablakot és megjeleníti azt a `sdl_init` és `SDL_RenderPresent` függvényekkel. Ez után a grafikus környezet egérrel való manipulálását írja le. Ez a while ciklusban valósul meg. A ciklus elején a program mindig kirajzolja az adott ciklusban a bábuk állását, lépési lehetőségeit (sárga négyzettel). Ez után az interakciót írja le, ahol kattintás vezérli az eseményeket. Egy kattintásra megnézi, hogy milyen bábúra kattintott a felhasználó és az adott bábu lépési lehetőségeit beírja a `sakk_hatter` tömbbe (`fekete_huszar_lep` és stb. függvényekkel). Aztán az első kattintás után a kiválasztva bool igazra állítódik és beleép egy másik ciklusrészre, ahol először megnézi, hogy ugyan arra a bábúra kattintott-e mert ha igen a kiválasztást elveti és `alap_hatter`t állítja vissza. Másodszor megnézi, hogy a kattintott mező az lépő mező és ha az akkor oda lép. A többi programrész az ablak bezárását és kattintás végét írja le.

A `fv.c` fájlban az első függvény az `alap_hatter`, ami paraméterként kap egy bool 8x8-as tömböt, majd annak minden elemét true-ra állítja. A második függvény az `alap_tabla`, ami paraméterként kap egy Babu típusú 8x8-as tömböt (A Babu típus enum-al van leírva a `fv.h` -fájlban) és azt a sakk szabály szerinti alaphelyzetére állítja. Majd a harmadik a `babu_rajzol` függvény, ami a paraméterként kap egy renderer pointert (ami a megjelenítő), egy hatter pointert (ami egy kép), egy sakk háttért, ami leírja ,hogy melyik négyzeten igaz és végül az x és y, amik leírják az adott helyet a `sakk_hatter` két dimenziós tömbben. Ezek után az x és y helyzetű négyzetet kirajzolják a képernyőre, ha igaz az adott helyen a `sakk_hatter` tömb. A negyedik függvény a `kep_betolt` függvény, ami paraméterként megkap egy megjelenítőt(`SDL_Renderer-t`) és egy kép helyét. A kapott információból betölti a kép helyén lévő helyeket. Az ötödik függvény a `tabla_rajzol`, ahol paraméterként megkapja a `SDL_Renderer-t` és a táblát (kép) és az ablakba kirajzolja azt. A hatodik függvény az `sdl_init`, ami széles és magas int alapján egy széles szer magas ablakot hoz létre arra a pontra ahova a `pwindow` pointer mutat, illetve létrehoz egy megjelenítőt, ahova a `prenderer` pointer mutat.

A lepes.c fájlban, minden függvény célja hogy az adott bábutípus lépését leírja, ennek elérésére kap paraméterben egy Babu típusú sakk_tabla tömböt, ami a bábuk elhelyezkedését adja meg, e-mellett kap egy sakk_lep tömböt, amin kiírja a lehetséges lépéseket és végül megkapja a vizsgált bábu elhelyezkedését. Mostani állapotukban nem a szabály szerint lépnek ezek a függvények.

A program lefordítása:

```
gcc main.c fv.c lepes.c -o out `sdl2-config --cflags --libs` -lSDL2_gfx -lSDL2_ttf -lSDL2_image -lSDL2_mixer
```

A program futtatása (linuxban):

```
./out
```