

Programozói Dokumentáció

Snake-játék

A program egy Snake típusú játék megvalósítása **C** nyelvben **SDL 2.0** felhasználásával. A program **CodeBlocks 16.01**-ben készült **GNU GCC** fordítóval. A program az alábbi könyvtárakat használja:

```
mingw32; SDL2main; SDL2; SDL2_ttf;
```

A program 4db programfájlt és 3db header fájlt tartalmaz:

A **main.c** fájl:

Ez a fájl felelős az SDL inicializálásáért és a menü kirajzolásáért.

A fájl tartalmaz egy **close_requested** nevű extern bool globális változót, mely akkor igaz, ha a felhasználó ki szeretne lépni és rákattint az X gombra. Ha ennek értéke igaz akkor a program kilép a fő menü ciklusból, felszabadítja a foglalt memóriát valamint kilép az SDL-ből.

A fájl a main függvényen kívül egy **InitWinRender** nevű bool függvényt mely Inicizalizálja az SDL-t megnyit egy ablakot és létrehozza a Renderer-t mellyel az egész program dolgozik.

A **main** függvény a menüt rajzolja ki és a teglalapban(deklarálva a jatek.c fájlban) nevű függvény segítségével eldönti, hogy melyik kirajzolt téglalapra kattintott a felhasználó, így hívva meg az adott függvényt.

A **jatek.h** fájl:

Ebben a fájlban található a több fájlban is használt extern változók, mint a már ismert **close_requested**, az **eger** változó, mely az eger pozíciójának meghatározásában használt.

A **kigyó** nevű láncolt lista a kígyó méretét és mozgását tárolja.

A **spawnpos** nevű struktúra a kígyók spawnolási pozícióit tárolja.

A fájl ezenkívül tartalmaz még 4 struktúrát, melyből az első a **mousepos** az eger x és y koordinátáját tárolja. A **TextColor** és a **RectData** struktúrát a **StructForTTF** struktúra tárolja, melyet a szövegek képernyőre való kiírásához használok.

A fájl többi eleme a jatek.c függvényeinek deklarációi.

A **jatek.c** fájl:

Ebben a fájlban fut a játék egy és kétjátékos módjának függvénye is, valamint az azokban használt legtöbb függvény is ebben található.(Kivételek pl.: fájlba írás/olvasás)

A fájlban található egy **meret** nevű extern int változó mely a kígyó és az alma méretének változtatására, használt, ezt egy settings.c-ben található függvény változtatja meg.

Az alma nevű SDL_Rect az **alma** pozícióját és méretét tárolja.

A **teglalapban** nevű függvény határozza meg, hogy az egér pozíciója egy bizonyos SDL_Rect téglalapon belül van-e, így valósítva meg a gombok funkcióit.

A **freeList** nevű függvény szabadítja fel kilépéskor a láncolt lista elemei által lefoglalt memóriát.

A **TTFCreateAndPrint** függvény a szövegek kiírását könnyíti meg a képernyőre. A függvény várja a renderert, melyre kiírja a szöveget, várja a nyomtatandó sztringet, a szöveg méretét, és egy StructForTTF típusú változót.

A **getInput** nevű függvény bekéri a felhasználótól a nevét (a konzolban), hogy frissítse vele a toplistát.

A **Ki_Nyert** nevű függvény csak kétjátékos mód esetén hívódik meg és eldönti, hogy ekkor melyik játékos nyert.

A **Nyert_e** függvény csak egyjátékos mód esetén hívódik meg és eldönti a minpont extern változó segítségével, hogy a játékos felkerül-e a toplistára. Felkerülés esetén meghívja a getInput és a dicsosegFrissit függvényt.

A **check4Clash** nevű függvény ellenőrzi, hogy magába/egymásba mentek-e a kígyók és eszerint tér vissza true vagy false értékkel. Egyjátékos mód esetén a függvény második argumentuma NULL kell, hogy legyen.

A **GameOver** függvény akkor jelenik meg ha a check4Clash visszatérési értéke true. Egyjátékos mód esetén itt is NULL legyen az utolsó argumentum. Egyjátékos mód esetén meghívja a Nyert_e függvényt, kétjátékos mód esetén a Ki_Nyert függvényt.

A **SnakeBodyLogic** függvény start=1 érték esetén létrehozza a kígyót majd az input függvényből kapott irány alapján megfelelő helyre helyezi annak a legújabb elemét. Ha a kígyó egy mezőre ér az almával akkor növeli a kígyót eggyel tér vissza, egyébként nullával. Ez alapján hívódik meg a SpawnApple függvény.

A **SpawnApple** függvény az almák spawnolásáért felelős. Egy játékos mód esetén ennek a 2. argumentumaként is NULL-ot kell megadni.

A **CreateGame** függvény dinamikusan hozza létre a pályát, mely habár kicsit lassabb, jövőben több fejlesztési lehetőséget enged meg, mintha nem dinamikusan foglalnám.

A **DrawPoints** függvény rajzolja ki a képernyőre a játékos pontszámát.

A **DrawOnScreen** függvény felelős a játékos grafikus megjelenítéséért, miközben fut a játék mindent ez rajzol ki, csak Game Over esetén nem.

Az **input** függvény az egyes számú játékos inputait kezeli, ez alapján állítja be a kígyó irányát.

Az **input2** dolga ugyanaz, mint az előző függvényé, de ez a második játékos inputait kezeli, ezáltal csak kétjátékos mód esetén is hívódik meg.

Az **egyjatekos** függvény az egyjátékos mód fő függvénye, ebben fut a játék ciklusa is, mely meghívja a fenti függvények nagyrészét.

A **ketjatekos** függvény dolga ugyanez, csak kétjátékos mód esetén, tehát bizonyos függvényeket kétszer is meghív, mindkét kígyóra egyszer.

A **dicsoseg.h** fájl:

Ez a fájl tartalmazza a **minpont** nevű extern változót, mely azt határozza meg, hogy a játékos felkerült-e a dicsőséglistára.

Egy **dicsoseg** típusú struktúrába olvasom be a fájlból beolvasott értékeket(helyezés, név, pontszám).

A **dicsoseg.c** fájl:

A top10 nevű **dicsoseg** típusú tömb tárolja a fájlból beolvasott értékeket.

A **fajl_be** függvény olvassa be a dicsoseg.txt-ből az eltárolt értékeket.

A **fajl_ki** írja ki a dicsoseg.txt-be a (frissített) dicsőség listát.

A **dicsosegFrissit** frissíti a dicsőséglistát a felhasználó nevével, helyezésével és pontszámával.

A **dicsosegKiir** függvény írja ki a képernyőre a dicsőséglistát ha a felhasználó rákattint erre a menüpontra a menüben.

A **settings.h** fájl:

A fájlban azt **meret** nevű extern int változó értékét lehet változtatni a beállításokban, mely a kígyó és az alma méretét változtatja.

A **settings.c** fájl:

A **MeretCsere** nevű függvény változtatja meg a **meret** nevű változó értékét a felhasználó inputja alapján, így indirekten növelve/csökkentve a játéktér méretét.

A **beallitas** függvény felel a beállítások menüpont grafikus részéért, azaz a méretezés gomb kirajzolásáért, valamint ez mondja meg a felhasználónak is, hogy a konzolba kell mennie azért, hogy megváltoztassa a méretet.