

Aknakereső

Programozói dokumentáció

A program egy aknakereső játék, ahol az $x*y$ nagyságú pályán vannak elszórva aknák, a játékosnak az a célja, hogy megtalálja az összes cellát, ahol nem található akna, úgy, hogy közben nem talál aknát. A játékban van 60 perces időlimit is, illetve egy 5 fős ranglista is az idő szerint rendezve, melyet egy fájlból olvas be és a végén ír ki a program. Futtatásához szükség van, az SDL2 grafikus könyvtárra.

A program több modulból épül fel.

Vezérlés modul (vezerles.h, vezerles.c):

A vezérlő modul tartalmazza a játék működéséhez szükséges függvényeket és típusokat. Itt készül el a játémező és az aknák kiosztása, cellák megjelölése, megmutatása. Ezt a modult fel lehet használni akár egy konzolos, akár egy grafikus játék elkészítéséhez is.

Enum all: Egy adott cella állapotára használt felsorolt típus. Lehet az értéke bomba, üres, jelölt bomba, jelölt üres, felfedett és robbant.

Struct cella: Egy cella állapotára és értékére (hány akna van körülötte) vonatkozó típus, a játékpálya ilyen típusú kétdimenziós tömb.

Struct pozicio: Egy sor, oszlop tárolására alkalmas struktúra.

Struct game: A játék fő adatait (pályára mutató pointer, magasság, szélesség, aknák száma, és a felfedett cellák száma, időtartama, játékos neve) tartalmazó struktúra.

Enum hibatipus: A játék elindulására vonatkozó típus, ha valami nem jó érték azt adja meg (meret, aknaszam, mindketto)

Hanyakna: Megállapítja, és visszaadja, hány db akna van a cella körül. Ezt úgy teszi meg, hogy a megadott pozicio körüli $3*3$ -as mátrixot járja be.

Palyainit: Egy kétdimenziós tömböt hoz létre, a kapott game struktúrájának $(magasság+2)*(szélesség+2)$ méretében. A $+2$ -re azért van szükség, mert így nem kell figyelni, hogy a hanyakna függvény esetleg túl- vagy alulindexeli a tömböt, ha a szélén áll a pályának. Lefoglalja a memóriában, ha ez nem lehetséges, akkor NULL értéket ad a pályának. Normál esetben a kért számú bombát szétosztja random a pályán, majd minden cella értékét beállítja. Visszatérési értéke a foglalás sikerességét mutatja.

Palyan: Megmondja, hogy a kapott pozicio a $magasság*szélesség$ méretű pályán van-e.

Felfed: A kapott pozicio struktúra helyen lévő állapotot változtatja felfedettre vagy robbantra a tartalmától függően a game típusú paraméterben, ha az nem volt épp megjelölve. Ha a mező értéke 0, akkor a környező cellákra is meghívja ezt a fgv.-t rekurzívan, ha azok pályán vannak és nincsenek már felfedve.

Megjelol: A kapott pozicio-ban lévő cellát jelöli meg, vagy tünteti el róla a jelölést, a cella állapotát megváltoztatva.

Felszabadit: A kapott game pályáját szabadítja fel a memóriából.

Nyerte: Megállapítja a felfedett cellák és a bombák számából, hogy megtalálta-e már a játékos az összes nem bomba mezőt.

Mehet: Fgv., ami megállapítja, hogy megfelelőek-e a game típusú paraméter értékei. Visszatérési értéke egy logikai változó, ami erre utal. Ha van hiba, akkor a hibatípus paramétert olyanra állítja be, amilyen a hiba.

Megjelenítő modul (megjelenites.h, megjelenites.c):

Ez a modul felel a program grafikájáért, hangjaiért és a különböző események (pl egérgomb lenyomása, ablak bezárása) végrehajtásáért.

Sdl_init: Ez az új InfoC SDL2 projekt nyitáskor létrejövő programból az SDL inicializáló függvény, mely inicializálja az SDL-t, létrehoz egy ablakot, illetve egy renderert.

Input_text: InfoC-ről vett SDL-es szövegbekérő melyben a kilépési feltétel az egér mozgására van megváltoztatva az enter lenyomásáról. <https://infoc.eet.bme.hu/sdl/>

Idoformatum: Ez a fgv. kap egy másodpercben megadott számot, és egy sztringet csinál PP:SS (pl: 05:21) formátumban. A sztringre mutató pointerrel tér vissza.

Fomenu: A játék induláskor megjelenő menüablakot hozza be. Visszatérési értéke, hogy a játékos elindítja-e, az új játékot. Ez egy képernyőre kreált gomb lenyomásával lehetséges. Paraméterei: egy game típusú pointer és egy listára mutató pointer. Itt lehet megadni a pálya magasságát és szélességét, az aknák számát és a játékos nevét (alap esetben ez Jatekos). Itt van kiírva a ranglistán szereplők ideje, neve, pályamérete és aknaszáma. A piros x-szel be lehet zárni.

Cellatrajzol: Megkapja a pályát, a renderert, és a cellaméretet, majd ezek alapján kirajzolja a képernyőre az egész pályát, a cella állapotai és értékei alapján.

Melyikcella: Megkap egy eseményt, egy pályát (ehhez csak a szélesség kell nekünk, és a cellaméretet, majd a kattintás helyéből megállapítja melyik cellára kattintottunk, és ezt visszaadja egy pozíció struktúrában.

Idozites: Az időzítéshez szükséges fgv., ami kreál egy user eventet kapott ms-onként.

Jatekablak: Azt az ablakot nyitja meg, ahol maga a játék folyik majd. Van egy új játék gomb, egy főmenü gomb, egy óra, ami másodpercenként változik, illetve maga a játékos. Bal gombbal lehet felfedni, jobbal megjelölni a cellákat. Paraméterként megkapja a játékot, egy új játék és egy menu logikai változó címét (tudjuk, hova akarunk menni az ablakból), illetve egy lista pointere cím szerint átadva.

Fájl- és listakezelő modul (fajlkezeles.h, fajlkezeles.c):

Ez a modul a fájlkezeléssel, és a ranglista műveleteinek kezelésével foglalkozik. A fájl formátuma olyan, hogy soronként van egy játék adata: elől egy egész szám (a játék másodperceinek száma) utána egy szóköz, majd „név magasságxszélesség aknaszám db”, amit be lehet egy sztringbe olvasni, mert csak kiírni kell ezeket a képernyőre.

Struct listaelem: Egy lista elemének típusa. Tartalmaz egy int típusú időt (másodpercek), egy következő listaelemre mutató pointer, és egy egyéb nevű sztringet, amiben a játékos neve, pálya magassága, szélessége, és az aknaszám van „név magasságxszélesség aknaszám db” (pl: Gabor 30x30 50 db) formában.

Listabeolvas: Megnyitja a ranglista.txt fájlt, majd beolvassa abból a listát. Ha nincs elem NULL pointerrel tér vissza, egyébként meg a lista első elemére mutató pointerrel.

Listafelszabadit: Felszabadítja a memóriából a lista által foglalt helyet.

Listakiir: A kapott listából kiírja a ranglista.txt nevű fájlba az első 5 elemet.

Listabarak: Cím szerint átadva megkapja a lista első elemére mutató pointert, és az aktuális játék adatait. Ebből elkészít egy listaelemet, majd idő szerint a megfelelő helyre teszi a listában.

Debugmalloc.h:

A dinamikus memóriakezelés ellenőrzését végző modul.

Main.c:

A main függvényt tartalmazza csak, mely meghívja a fájlkezelős, megjelenítő és pályainicializáló függvényeket