

A PROGRAMOZÁS ALAPJAI 2.

VÉGLEGESÍTETT HÁZI FELADAT SPECIFIKÁCIÓ

ROLEPLAY GAME

KÉSZÍTETTE: KOMONYI ORBÁN BALÁZS, HWQMMR
komonyibalazs@gmail.com

2025. 04. 25.

TARTALOMJEGYZÉK

Informális házi feladat leírás.....	3
Formális házi feladat leírás	3
Az elkészítendő szoftver célja	3
Elvárások	3
Elvárt bemenet	3
Elvárt kimenet.....	3
Implementálandó funkciók	3
Karakter osztályok:	3
Fegyverrendszer:.....	3
Körökre osztott harcrendszer:	3
Felhasználói interakciók kezelése:	3
Szöveges visszajelzések:.....	3
Játék szerkezete és működése	4
Hibatűrés és robusztusság	4
Tesztelési követelmények	4
Felhasználói interakció	4
Fejlesztéshez használt technológiák	4
Egyéb követelmények.....	4
Futtató környezet követelményei.....	4
Célközönség	4

Informális házi feladat leírás

Egy szöveges, konzolos szerepjátékot készítünk, amelyben a felhasználó különböző karaktertípusok közül választhat (harcos, varázsló, íjász), majd körökre osztott harcban küzdhet meg automatikusan generált ellenfelekkel. A játék során a karakterek tapasztalati pontokat szereznek, szintet lépnek, fegyvereket használnak, és speciális akciókat (pl. menekülés) is végrehajthatnak.

Formális házi feladat leírás

Az elkészítendő szoftver célja

A cél egy turn-based RPG harcrendszer megvalósítása C++ nyelven, amely polimorfizmust használ a karakterek viselkedésének kezelésére. A játékban a karakterek saját támadó logikával rendelkeznek, és a harcot egy külön Combat osztály koordinálja.

Elvárások

Elvárt bemenet

A program konzolos menüben kínál karakterválasztást (pl. Warrior, Wizard, Archer).

A játékos körönként dönthet az akcióról: támadás, regeneráció, fegyver javítása/újratöltése, fegyvercsere, információ lekérdezése vagy menekülés.

Az ellenfél automatikusan generált karakter, szintje igazodik a játékoshoz.

Elvárt kimenet

A harci körök eredményei (támadások, sebzések, gyógyulások, fegyver javítása, újratöltése, győzelem/vereség, szintlépés) szövegesen jelennek meg a konzolon.

A karakterek állapota (életerő, tapasztalat, mana, pajzs, szint, fegyverek) dinamikusan frissül a játék során, ezek bármikor lekérdezhetőek a menüből.

Implementálandó funkciók

Karakter osztályok:

Harcos, Varázsló, Íjász. Mindegyiknek egyedi tulajdonságai (pl. pajzs, mana, lőszer) és fejlődési logikája van.

Fegyverrendszer:

- Közelharc fegyverek (kard, dárda, balta): tartósság, élettartam, javíthatóság.
- Mágikus fegyverek (varázspálca, villám bot, tűz pálca, jég pálca): mana költség, mágikus sebzés.
- Távolsági fegyverek (íj, számszeríj, hosszú íj): lőszer, újratöltés.
- Minden karakter egyszerre több fegyvert is birtokolhat (max. 3), fegyvercsere lehetséges.

Körökre osztott harcrendszer:

- Akcióválasztás: támadás, gyógyulás/regeneráció, fegyver javítása/újratöltése, menekülés, fegyverváltás, információk megtekintése, kör kihagyása.
- Szintlépés és karakterfejlődés támadásonként szerzett tapasztalat alapján.

Felhasználói interakciók kezelése:

- Konzolos menürendszer (főmenü, harcménü, információs menü, inventory menü).
- Felhasználói bemenet ellenőrzése, hibakezelés (érvénytelen választásnál visszajelzés).

Szöveges visszajelzések:

- Minden fontos eseményről (sebzés, gyógyulás, fegyvercsere, győzelem, vereség, szintlépés stb.) a konzolon történő kiírás.

Játék szerkezete és működése

A játékos a program indításakor nevet adhat a karakterének, majd kiválaszthatja a karakterosztályt (Warrior, Wizard, Archer). Minden karakter alapértelmezetten kap egy kezdő fegyvert (kard, varázspálca vagy íj), amely szintfüggő tulajdonságokkal is rendelkezhet.

A főmenüben a játékos a következőket teheti:

- Kóborol a vadonban (random ellenséggel harcolhat)
- Regenerálódik/javítja vagy újratölti fegyverét (osztályfüggő módon)
- Fegyvert vált (ha legalább 2 fegyvere van)
- Megnézi karaktere vagy fegyverei/inventoryja adatait
- Kiléphet a játékból

Harc során a játékos minden körben dönthet a fenti akciók egyikéről.

Az ellenfél karakter automatikusan generálódik (szint, osztály, fegyver).

Tapasztalatot szerezni ellenség legyőzésével lehet, szintlépés automatikusan megtörténik.

Szintlépéskor a karakter maximális életeréje/mana/pajzs nő.

A karakterek és fegyverek minden tulajdonsága (HP, XP, szint, fegyver neve/sebzés/tartósság/mana/lőszer) bármikor lekérdezhető, szövegesen megjelenik.

Hibatűrés és robusztusság

Minden hibás vagy érvénytelen bemenetnél szöveges hibaüzenet jelenik meg.

A rendszer gondoskodik arról, hogy a fegyver-inventory ne lépje túl a maximális méretet, illetve hogy csak létező fegyverre lehessen váltani vagy cserélni

A fegyverek használata, javítása, töltése csak akkor lehetséges, ha az adott fegyver igényli (pl. nem teljesen töltött, nem teljesen javított stb.).

A játék során bekövetkező kivételek (std::invalid_argument, std::out_of_range stb.) mindenhol kezelve vannak, részletes hibaüzenetekkel.

Tesztelési követelmények

Minden karakterosztály, fegyvertípus, harci akció, menüpont működése kipróbált, hibás bemenetek ellen védett, program nem omlik össze hibás input esetén sem.

Felhasználói interakció

A program szöveges menürendszerrel irányítja a játékot. A játékos karaktert választ, majd minden körben dönthet az akciójáról. A harc automatikusan lezajlik a kiválasztott akciók alapján, az ellenfél lépései automatikusak. Bármikor lekérdezhető a karakter vagy az ellenség aktuális állapota.

Fejlesztéshez használt technológiák

- A játékot C++11 szabvány szerint készítem, Visual Studio 2022 fejlesztőkörnyezetben.
- Az implementáció során modern C++ eszközöket (std::vector, std::unique_ptr, dynamic_cast, static_cast, polimorfizmus) alkalmazok.
- Verziókezelésre GitHubot használlok.

Egyéb követelmények

Futtató környezet követelményei

A program Windows operációs rendszer alatt fut Visual Studio 2022 környezetben.

A system("cls") metódus használata csak Windows alatt támogatott.

Célközönség

A programom az RPG játékok kedvelőinek készült.