**Féléves feladat**

*Tűzeset Szimuláció*

Szoftverfejlesztés Párhuzamos Architektúrákra

Mérnökinformatikus, Msc

*Készítette:*

Liszkai Keve Áron

T0BQQL

# Célkitűzés

Nem titkolt célom az, hogy a mesterképzés színvonalának megfelelő minőségű munkát adjak be a fél éves otthoni munkám végeredményeképpen. Ezért igyekezni fogok minőségi munkát kiadni a kezemből, hogy az minél magasabb érdemjegyben részesüljön.

# Projekt témája

A párhuzamos programozás témakörében való kisebb internetes kutatásom során két témát találtam, amelyek engem komolyabban megmozgattak.

Az egyik ilyen témakör a különböző szimulációk témaköre. Sok internetes forrás ajánlotta a jól párhuzamosítható példaprogramok között a különböző gázok/folyadékok, és más fizikai jelenségek szimulációját (pl.: Gravitációs N-test probléma). Az általam érdekeseknek vélt projektek főleg a fizikai valóság lemodellezésén alapuló, és jól párhuzamosítható feladatokra épülő programok.

A másik téma, amit érdekesnek találtam, az a különböző (akár számítógépes játékokból is ismerhető) „mesterséges intelligenciák” (AI) párhuzamos kezelése. Természetesen itt egyszerűbb mesterséges intelligenciákról van szó (bár találtam forrást, ami egészen a tanuló algoritmusokig cifrázza a dolgot), itt a hangsúly a párhuzamosíthatóságon van, hiszen minden egyes egyed (bizonyos korlátokon belül) rendelkezhet akár saját szállal, így a teljesítmény egyértelműen növelhető.

A fent említett két téma összefűzéséből született meg a projekt, amivel a félév során szeretnék foglalkozni, és úgy gondolom, hogy megfelel az elvárt szintnek.

# Feladat

Adott egy fizikai, kettő dimenziós, zárt tér (pl. iroda). A térben jelen vannak bizonyos emberek. Az adott térben kialakul egy tűzeset. A feladat szimulálni a tűz terjedését, valamint a térben lévő emberek menekülését, természetesen mindezt grafikusan is ábrázolva. Végezetül elkészíteni ennek a több szálas, valamint az egy szálas megoldását, és azokat összevetni teljesítmény szempontjából.

A feladatnak három fő alkotó része van: játéktér, ember, tűz. A későbbiekben ez a három alkotóelem lesz bemutatva. A feladat bonyolultsága nagyban függ a megvalósítás minőségétől, amely a ráfordított idővel arányos. Hiszen, ha csak arra gondolunk, hogy a fent említett három szereplő bármelyike akár lehet random generált is, de lehet beégetett is. Az egyes emberek akár rendelkezhetnek bonyolultabb pathfinding algoritmussal is, de akár még egyéni jellemzőkkel, viselkedési formákkal is. Természetesen ezek inkább a szimuláció életszerűségét növelik.

# Játéktér

A szimuláció statikus tagja. Rendelkezik legalább egy kijárattal. Rendelkezik falakkal, amik az emberek számára átjárhatatlanok, a tűz számára lassító tényező. A játéktér szobákra osztható, amelyek rendelkeznek egy, vagy több ajtóval (egyszerű pathfinding esetén ezek az ajtók lesznek a menekülési útvonal pontjai).

# Tűz

A tűznek egy kiindulási pontja van, a tűzfészek. A szimuláció paraméterei szerint lehet egy, vagy több darab, lehet random, vagy egérkattintás helyén. A tűz az epicentrumtól kifele mutató irányba, egy állandó sebességgel táguló kör. Ha a tűz útjába fal kerül, akkor az egy bizonyos időre lelassítja a tüzet, és természetesen innentől kezdve megváltozik a geometriai alakja. A tűz egyik fő tulajdonsága, hogyha elér egy embert, akkor az adott ember megég, és kiesik a szimulációból.

# Ember

A legérdekesebb, és leginkább számolásigényes tagja a szimulációnak. Lehet egy, vagy több, elhelyezkedésük lehet random, vagy egérrel elhelyezett. Az ember rendelkezik egy látótérrel, amin belül képes érzékelni a tüzet. Az emberek alapesetben random mozgást végeznek, vagy egy helyben állnak (ezeket az állapotokat akár váltogathatják is), egészen addig, amíg meg nem látják a tüzet. Miután észlelték a tüzet, a játéktér kijárata felé veszik az irányt (ez persze lehet egy bonyolultabb, vagy akár egy egyszerűbb algoritmus alapján). Amelyik ember elérte a kijáratot, az megmenekült.

Az emberek kapcsán megemlíthető esetleges színesítésként még elképzelhető akár az is, hogy az emberek képesek legyenek figyelmeztetni egymást, ha már látták a tüzet. Vagy akár még az is, hogy bizonyos emberek, meg direkt ne csinálják ezt.

# Füst

Esetleg plusz tényezőként a füst még felsorolható, mint nehezítő elem. Ezt nem tartozik szorosan szimulációhoz, azonban egy jó példa arra, hogy a szimuláció összetettsége még sok módon növelhető. A füst is egy érdekes résztvevő lehet, amely hasonlóan viselkedik a két dimenziós térben, mint a tűz, azzal a két különbséggel, hogy gyorsabb, valamint a falak számára áthatolhatatlanok.

# Egyszálas vázlat

Az egyszálas változat folyamán az egyes komponensekkel kapcsolatos műveletek (pl.: mozgatás) egymás után hajtódnak végre egyetlen közös szálban. Logikai vázlatát a következő ábra mutatja:

Ember 1 mozgatás

Ember N mozgatás

Tűz mozgatás

Füst mozgatás

Játéktér generálás

SZÁL 1

# Több szálas vázlat

A többszálas megoldás során a bevárásos szinkronizáció módszerét fogom alkalmazni. Azaz, egy főszál első lépésként legenerálja a játékteret, majd létrehoz további szálakat, amelyek mindegyik egy komponensért felelős (Ember, Tűz). Az egyes komponensek Kiszámolják a hozzájuk tartozó objektumok jövőbeli koordinátáit, valamint egyéb értékeit, és utána egy randevú ponton találkoznak. Itt a főszál a kapott koordináták segítségével frissíti a képernyőt, így a mozgás létrejön.

**Randevú pont**

Ember 1 mozgatás

Ember N mozgatás

Játéktér generálás

Tűz mozgatás

Füst mozgatás

Képernyő frissítése

Szál 1

Szál N

Szál N+1

Szál N+2

# Időmérés, összehasonlítás

Mind a két megoldás során a teljesítmény, és az idő folyamatos mérés alatt lesz, így a két szimuláció futtatása után az eredmények egyértelműen összehasonlíthatóak lesznek. Természetesen több szimuláció adatait fogom átlagolni a pontosabb végeredmény érdekében.

# Grafikus Interfész (GUI)

A grafikus megjelenéshez a Visual Studio-ban megtalálható Windows Forms Application-t fogom alkalmazni. Ez képes ablakos alkalmazások létrehozására, valamint azokon objektumok mozgatására. Úgy gondolom, hogy a projekt igényeinek eleget tud tenni.

# Technológiák

A projekthez a Visual Studio 2017 nevű fejlesztőkörnyezetet fogom használni a .NET keretrendszerrel. A projekt fejlődése nyomon követhető a következő nyilvános GitHub linken: [tuzeset-szimulacio](https://github.com/keveliszkai/tuzeset-szimulacio).

# Verziók

Természetesen, mivel a projekt nehézségének megbecslése rizikós, ezért több lehetséges verzió képzelhető el a végeredmény tekintetében, így elképzelhető, hogy a jelen dokumentum nem lesz teljes mértékben összhangban a végeredménnyel.