

PROGRAMLAMA LABORATUVARI-I SAVAŞ SİMÜLASYONU OYUNU

Sevilay Üngör - Aslı Uzar
Bilgisayar Mühendisliği
Kocaeli Üniversitesi

I. ÖZET

Bu proje İnsan İmparatorluğu ve Ork Lejyonu isimli iki ırkın ordularının savaştığı bir strateji simülasyonunun gelişmesini amaçlamaktadır.

Savaş ızgara tabanlı bir alanda gerçekleşir. Oyunda yer alan canavarlar ve kahramanlar karşı tarafın birliklerini etkileyecek güçlere sahiptir; aynı zamanda araştırma seviyeleri, sağlık, yorgunluk ve saldırı-savunma etkenleri de ele alınarak savaşın sonucuna varılmıştır.

Projede JSON dosyaları istenildiği gibi manuel olarak okutulmuş, verilen url'ler koda entegre edilmiş dinamik olarak indirilmiş indirilen_senaryo.json'a kaydedilmiş, savaş mekaniği indirilen bu senaryo üzerinden yürütülmüş ve

mekanığın adımları savaşı_sim.txt'ye yazdırılmıştır.

Görselleştirme, istenilen üzere başlangıç ve sonuçları içermektedir.

II. GİRİŞ

Günümüz oyun endüstrisinde, strateji oyunları oyunculara taktiksel düşünme, planlama ve kaynak yönetimi gibi yetenekler kazandırarak önemli bir yer tutmaktadır. Bu projede, İnsan İmparatorluğu ve Ork Lejyonu adındaki iki farklı ırkın, birbirleriyle savaştığı bir strateji oyun simülasyonu geliştirilmiştir. Projenin amacı, iki ordunun birbirine karşı savaşını, ızgara tabanlı bir haritada gerçekleştirmek ve savaşın seyrini etkileyen çeşitli faktörleri simüle etmektir.

Proje, kahramanlar ve canavarların birimlere sağladığı avantajlar ve araştırma seviyelerinin savaş üzerindeki etkilerini de içeren karmaşık bir savaş mekaniğine sahiptir. Kahramanlar belirli birimlere bonus sağlarken,

canavarlar dost birliklere bonus sağlayarak savaşın akışına müdahale etmektedir. Her iki tarafın araştırma seviyeleri de savaşın sonucuna önemli bir etki yapar.

Görselleştirme, dış kaynaklı JSON dosyalarından gelen verilerle çalışır ve bu dosyalar hazır json kütüphanesi

kullanarak değil, C dilinde manuel olarak okunup işlenmiştir.

Birimlerin ,kahramanların,canavarların ve araştırma seviyelerinin özelliklerini içeren verileri uygun struct yapısı kullanarak eşleştirilmiştir, programda bu verilere uygun hesaplamaları yapılmıştır. JSON dosyalarındaki verilerin sıralarının ve değerlerinin sunum sırasında değiştirileceği nedeniyle, programın verilerin sırasına bağlı kalmaksızın doğru şekilde çalışmasına özen gösterilmiştir. Ayrıca, dosyalarda eksik veriler bulunması durumunda, program varsayılan değerler kullanarak eksik bilgileri tamamlayabilmesi sağlanmıştır.

Projenin bir diğer önemli unsuru, savaşın başlangıç durumu ve sonunda kalan birimlerin görselleştirilmesidir. Bu projede, ilk durumda seçilen senaryodan işlenen verilerle birlikte savaşa hazır birliklerin ,kahramanların ve canavarların konumunu gösterildikten sonra savaş sonuna kalan insan imparatorluğu ya da ork Lejyonun son durumu görselleştirilmiştir .

Proje, istenilen şartlara uygun olarak tamamen C programlama dili kullanılarak geliştirilmiştir. Savaş simülasyonunun yürütülmesi için gerekli veriler,

senaryoların internet üzerinden C dilinin cURL kütüphanesini kullanarak indirilip ayrıştırılması ve buna bağlı olarak savaş mekaniğinin oluşturulmasıyla elde edilmiştir.

III. YÖNTEM

Bu projede, İnsan İmparatorluğu ve Ork Lejyonu arasındaki savaş simülasyonunu gerçekleştirmek için aşağıdaki adımlar izlenmiştir:

1. Veri Toplama ve Hazırlama:

- a. Savaş simülasyonu için gerekli veriler, JSON formatında bizimle paylaşılan unit_types.json, research json, heroes.json, ve creatures.jsondan dosyalarından çekilmiştir. JSON dosyaları, manuel olarak C dilinde işlenmiş ve program içinde uygun structlar yapılarına dönüştürülmüştür. Verilerin sırasının değişken olabileceği göz önünde bulundurularak,doğru algoritmayla eşleştirmesi sağlanmıştır.
- b. Senaryoların cURL kütüphanesi kullanılarak dinamik olarak indirilip indirilen_senaryo.jsona kaydedilmesi ardından bu json dosyasının işlenmesiyle savaş mekaniğini kurgulamak mümkün olmuştur.

2. Savaş Mekaniği Tasarımı:

- a. Savaşın dinamiklerini belirlemek amacıyla, kahramanların ve canavarların sağladığı bonuslar dikkate alınmıştır. Her bir birimin sağlık, yorgunluk, saldırı ve savunma değerleri gibi etkenler ve araştırma seviyeleri de savaş mekaniğinin kurgulanmasında önemli rol oynayan kritik unsurlardır.
- b. Proje föyünde belirtilen hesaplama adımları göz önünde bulundurularak adım adım hesaplama yapılmış ,toplam savunma ,toplam saldırı, net hasar, net hasarın birimler üzerindeki hasar dağılımı ve güncel sağlık degerleri hesaplanmıştır ve bir ırk tamamen yok oluncaya kadar savaş devam etmekte olup kazanan taraf bulunmaktadır.

Ardından bizden istenilen hesaplama sonuçları c diliyle manuel olarak savaş_sim.txt dosyasına adım adım kaydedilmiştir.

3. Simülasyon Süreci:

- a. İndirilen senaryo dosyaları, program tarafından okunmuş ve savaş mekaniği bu veriler üzerinden çalıştırılmıştır. Savaşın adımları, savaş_sim.txt dosyasına yazdırılarak dökümanite edilmiştir.
- b. Program, eksik verilerle karşılaşılması durumunda bizden istenilen doldurma örneğin birimlerden süvari yoksa değeri 0 alınmalıdır gibi tasarlanmıştır.

4. Görselleştirme:

- a. Savaşın başlangıç durumu ve bitiş durumu c dilindeki Raylib.h Kütüphanesi yardımıyla görselleştirilmiştir. Bu görselleştirme, seçilen senaryonun başlangıç verileri ile birimlerin, kahramanların ve canavarların konumlarını göstermektedir. Birimlerin sayıları proje föyünde de belirtildiği gibi her bir ızgara için maksimum 100 birim olacak şekilde ayarlanmış ,taşma durumunda diğer kutucuğa geçiş yapılp toplam birim sayısının doğru şekilde görselleştirmesi sağlanmıştır. Herbir ırkın birimlerini ayır edebilecek şekilde görseller eklenmiş ayrıca birimlerin sağlığının doğru şekilde görselleştirildiği can barı konulmuştur. Bir birliğin sağlığının tamamen bitmesiyle savaştan çıkmış tüm birimleri tükenen ırk savaşı kaybetmiş son görselleştirilmeye dahil edilmemiş kaybeden ırkın kahraman ve canavarı hariç bunun sebebiyse kahraman ve canavarın cana sahip olmamasıdır .Galip taraf ise savaşın sonucunda kalan birimleri ve kahramanları ve canavarlarıyla birlikte kullanıcıya sunulmuştur.

IV. DENEYSEL SONUÇLAR

A. Deneyin veya Simülasyonun Tanımı

Bu çalışmada, İnsan İmparatorluğu ve Ork Lejyonu isimli iki farklı ırkın savaştığı bir strateji oyunu simülasyonu geliştirilmiştir. Simülasyon, C programlama dili kullanılarak yazılmış olup, savaş mekaniklerinin işleyişini ve sonuçlarını analiz etmek amacıyla belirli senaryolar kullanılarak gerçekleştirilmiştir.

B. Simülasyon Yazılımı

Simülasyon, aşağıdaki yöntemler ve araçlarla yürütülmüştür:

- **C Programlama Dili:** Proje, tamamen C dilinde geliştirilmiştir.
- **JSON Verileri:** Senaryo verileri JSON formatında tanımlanmış ve manuel olarak ayrıştırma teknikleri kullanılarak işlenmiştir. Veriler, cURL kütüphanesi aracılığıyla internetten indirilmiştir.
- **Birim Parametreleri:** Savaş parametreleri (saldırı gücü, savunma gücü, sağlık durumu, kritik vuruş oranı vb.) program tarafından işlenmiştir.

C. Test Senaryoları

Farklı savaş senaryoları, JSON dosyaları aracılığıyla programa sağlanmış ve bu dosyalarda her iki tarafın başlangıç birimleri, kahraman ve canavar bonusları ile araştırma seviyeleri tanımlanmıştır:

- **unit_types.json:** İnsan ve Ork birimlerinin saldırı, savunma ve sağlık gibi özelliklerini içerir.
- **heroes.json:** Kahramanların hangi birimlere nasıl bonuslar sağladığını belirtir.
- **creatures.json:** Canavarların savaş sırasında sağladığı avantajlar ve düşmana verdikleri dezavantajları tanımlar.
- **research.json:** Araştırma seviyelerinin kritik vuruş ve savunma güçlerine etkisini içerir.

D. Test Koşulları ve Parametreler

- **Senaryo Başlangıcı:** Simülasyon, cURL kütüphanesi yardımıyla indirilen senaryolarda başlar. Ve Raylib kütüphanesiyle beraber görselleştirilir.
- **Birim ve Kahraman Testi:** Her senaryoda farklı sayılarda ve türlerde birimlerin savaştığı durumlar incelenmiş, kahramanlar ve canavarların etkisi test edilmiştir.

Örnek Senaryo:

Proje sayfasında paylaşılan 1.senaryo dosyasının içeriği

- İnsan İmparatorluğu, 300 piyade ve 200 süvari birimi, kahraman olarak Yavuz sultan selim, canavar olarak tepegöz ve araştırma seviyesi 2 verilmiştir.
- Ork Lejyonu'nunsa 500 orkdövüşçüsü ve 400 mızrakçı, kahraman olarak Vrog kafakiran canavar olaraksa ateş iblisi verilmiştir.

E. Simülasyon Akışı

- Simülasyon başladığında, föyde de belirtildiği gibi her zaman insan imparatorluğu saldırı adımıyla savaşı başlatır mekanik hesaplamalar bir taraf malup olana kadar devam eder ,birimlerini sırayla karşılaştırarak saldırı ve savunma güçlerini hesaplar.
- Kritik vuruşlar, yorgunluk , kahraman ve canavar bonusları, araştırma seviyesi ve birim sayıları dikkate alınarak toplam saldırı ve savunma gücü hesaplanır. Föyde belirtilen net hasar denkleminde beraber her adımda net hasar hesaplaması yapılır.
- Ardından savunma gücüne oranla her bir birim için hasar dağılımı yapılır.
- Mevcut adımda hasar gören birimlerin sağlık durumu birim başına düşen hasar dağılımıyla beraber yeni sağlık hesaplamasıyla güncellenir ve sağlığı biten birimler sayıları ne olursa olsun savaş dışı kabul edilir ve artık saldırı ve savunmada yer alamaz.
- Simülasyon, taraflardan birinin tüm birimlerinin sağlığının 0 olmasıyla sona erer.

- Aciklanan bütün bu adımlar ,birimlerin sayisi, guncel sađlık deđerleri,net hasar dađilimi savas sim.txt dosyasına kaydedilir.

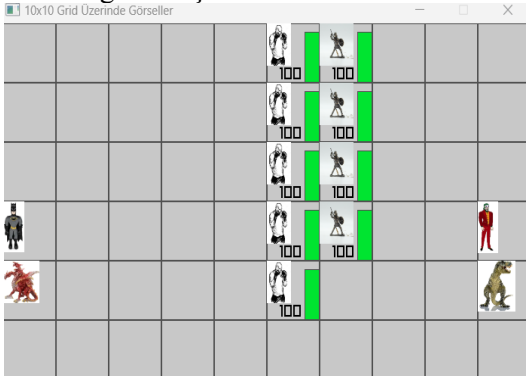
```
8. Adım:
toplam Saldırı Sonucu: 42600.00
toplam Savunma Sonucu: 7200.00
net hasar insan: 35400.00
okcu sayisi 0.00
piyade sayisi 0.00
suvari sayisi 200.00
kusatma sayisi 0.00
hasar dađilimi piyade : 0.00
hasar dađilimi okcu : 0.00
hasar dađilimi suvari : 35400.00
hasar dađilimi kusatma : 0.00
sađlık piyade : 0.00
sađlık okcu : 0.00
sađlık suvari : 0.00
sađlık kusatma : 0.00

İnsan ırkından hiçbir birim kalmadı.ork kazandı
```

- Savasin 1. senaryoya göre ilk durumunu görselleştirilmesi



- Savasin 1. senaryoya göre son durumunu görselleştirilmesi



V. SONUÇ

Bu çalışma, C dili kullanılarak geliştirilen bir strateji oyunu simülasyonu ile İnsan İmparatorluğu ve Ork Lejyonu arasındaki savaş mekaniklerini analiz etmektedir. Senaryoda verilen bonusların savaşın sonucunu etkilediđi, yorgunluđun performansı düşürdüđü ve sađlığın temelde adım sayısı ve savaşın bitiminde etkili olduđunu görselleştirmedeki başlangıç ve son karşılaştırmasıyla kullanıcıya açık bir şekilde sunmaktadır. Sonuç olarak, simülasyon taraflar

arasındaki savaşları objektif olarak analiz etmeyi sađlamış, benzer araştırmalara referans niteliđi taşımakta ve gelecekte daha karmaşık unsurlar eklenerek çeşitlendirilebileceđi vurgulanmaktadır.

VI. YAZAR KATKILARI

- Sevilay Üngör :
- Unit_types.jsonun c dilinde manuel olarak ayrıştırılması,
- cURL ile senaryoların dinamik olarak indirilip verilerin indirilen_senaryo.jsona yazdırılması ve C dilinde manuel olarak ayrıştırılması,
- Senaryodan çekilen verilerle beraber savaş mekanizmasının kurgulanması ve savas_sim.txtye yazdırılması
- Raylib.h ile sađlık barı,pngler,birim sayılarının eklenmesi ilk ve son durumun görselleştirilmesi
- Raporun yazılması

- Aslı Uzar :
- Research.json , heroes.json ve creatures.jsonun c dilinde manuel olarak ayrıştırılması,
- Görselleştirmenin yönteminin araştırılması , hangi kütüphanenin kullanılacađının saptanması
- Rapor yazılması
-

KAYNAKÇA

- [1] cURL, “cURL Documentation,” [Online]. Available: <https://curl.se/docs/>. [Accessed: Oct. 29, 2024]
- [1]raylib, “raylib - A C library to learn game programming,” [Online]. Available: <https://www.raylib.com/>. [Accessed: Oct. 29, 2024].
- [2]W3Schools, “How to Parse JSON in C,” [Online]. Available: https://www.w3schools.com/c/c_json.asp. [Accessed: Oct. 29, 2024].
- [3]Douglas Crockford. (2006). *JSON: The Fat-Free Alternative to XML*. Retrieved from <https://www.json.org/>

[4]Zeldman, J. (2010). *Game Programming in C++*.
Retrieved from
<https://www.gamedevelopment.tutsplus.com/>