

PROJE 1: Altın Toplama Oyunu

Bariş Arslan 180202112 – Melih Yeşilyurt 180202060

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

barisaarslan98@hotmail.com – melih58yesilyurt@outlook.com

1-) Özet

Geliştirdiğimiz oyunda oyunun özelliklerinin değerleri, oyun açıldıktan sonra kullanıcıdan isteniyor. Kullanıcı girmek istediği değerleri girdikten sonra oyun başlatılıyor. Oyun açıldıktan sonra oyun haritası ve oyuncuların puan durumları gözüküyor. Oyunda 4 oyuncu var. Bu oyuncuların özellikleri birbirlerinden farklı. Oyunda ayrıca gizli altınlarda bulunmakta. Bu gizli altınların açılması için üstlerinden geçilmesi lazım ya da C oyuncusunun özel gücünü kullanarak açması lazım. Tur atla tuşu sayesinde oyun oynanıyor. Oyunun bitebilmesi için ya haritadaki altınların bitmiş olması ya da sadece bir oyuncunun hayatta kalması gerekiyor. Oyun bittikten sonra, oyuncuların gittiği konumların listeleri ve oyunun özeti .txt uzantılı dosyalara çıktı olarak veriliyor.

2-) Giriş

Oyunun girişinde kullanıcıdan; oyun tahtasının genişliği, oyun tahtasının yüksekliği, altın bulunma oranı, gizli altın bulunma oranı, başlangıç altın miktarı, bir turdaki maksimum adım sayısı, A B C D oyuncularının hedef belirleme maliyeti ve hamle maliyeti, C oyuncusu gizli altın açma miktarı gibi değişkenlerin sayısı alınmaktadır. Bu değerler girildikten sonra oyunu başlat

butonuyla, oyun haritası açılmaktadır. Oyundaki karelerin hepsi butonlardan oluşmakta, oyundaki oyuncuların hareketleri butonların renginin değiştirilmesiyle çalışıyor. Oyunda A, B, C, D oyuncuları var. A oyuncusu en yakınındaki altını alıyor. B oyuncusu en karlı altını alıyor. C altını ilk önce belli bir miktar gizli altını açıyor sonra en karlı altını alıyor. D oyuncusu ise diğer oyuncuların hedeflerini görüp en karlı altını alıyor. Oyunun oynanması için tur atla butonuna basmak yeterli oluyor. Oyunun özeti ve oyuncuların nerelere gittiği dosyalara yazdırılıyor.

3-) Yöntem

C# dilinde bulunan Windows Form uygulamasıyla ara yüzü tanımladık. Bulunmayan fonksiyonlar ekibimiz tarafından yapıldı. Proje yapılırken birçok problemle karşılaşıldı. Problemlerin çözümünde ekip halinde çözüm aranıldı, aynı problem veya benzeri bir problemle karşılaşmış arkadaşlarımızla görüş alışverişi yapıldı, internette daha önce bu problemlerle karşılaşan insanların bu problemlerin üstesinden nasıl geldiği araştırıldı ve farklı bakış açıları kazanıldı. Bulunan birçok farklı çözümünden problemlerimizin çözümlerine uygun olanları seçildi.

Oyun ilk açıldığında kullanıcının karşısına ana menü geliyor. Kullanıcı ana menüdeki textboxlara değerleri girdikten sonra oyunu başlat butonuna basmasıyla oyun başlıyor ve oyun ekranı açılıyor. Oyun ekranı açılırken ilk önce Game_Load fonksiyonu çalışıyor. Bu fonksiyon baştaki ayarlamaları yapıyor. Haritadaki altınları tutabilmek için matris oluşturduk. İlk önce bu matrisin içine random bir şekilde gizli altınları yerleştirdik. Matristeki gizli altınların değerleri -1 olarak kaydediliyor. Gizli altınları yerleştirdikten sonra normal altınları yerleştirdik. Normal altınları yerlerine yerleştirirken değerlerini rastgele bir şekilde 5,10,15 veya 20 olacak şekilde yerleştirdik. Eğer o bölgede hiç altın veya gizli altın yoksa değeri 0. Oyun haritasındaki her kare butonlardan oluşuyor. Butonlar yazdığımız fonksiyon sayesinde oluşturulup ekrana bastırılıyor. Butonlar yerleştirildikten sonra oyuncuların renklerine göre haritanın köşeleri boyanıyor ve oyuncuların konumları positionA gibi dizilerde tutuluyor. Tur atla butonu sayesinde oyunun ilerlemesi sağlanıyor. Tourmanagement değişkeni sayesinde sıranın hangi oyuncu da olduğunu belirliyor. Sıra hangi oyuncudaysa onun fonksiyonunu çalıştırıyor ve altın miktarı yazısını güncelliyor. Butona her basıldığında ayrıca oyunun bitip bitmediğini kontrol ediyor. GameOver isimli bool değişkeni sayesinde kontrol ediliyor. İlk önce sahnedeki altınları kontrol ediyor. Eğer altınlar bittiyse GameOver True oluyor. Eğer bu kontrolden false olarak ayrıldıysa, Oyuncuların hayatta kalıp kalmadığını

kontrol ediyor. Eğer oyun bittiyse, Altın miktarlarını karşılaştırıp birinciyi ekranda gösteriyor. Sonrasında ise oyunun özetini dosyaya çıktı olarak veriyor.

Sıra A oyuncusuna geldiği zaman ilk önce hedef belirleme özelliği çalışıyor. Eğer hedefleri X:0 Y:0 şeklinde ise veya hedefinin bulunduğu yerdeki altın alındıysa hedef belirliyor. Hedef belirlerken hedef belirleme ücreti kasasından azaltılıyor. En kısa yolu belirlemek için mesafe matrisi oluşturduk. Bu matrisin içerisinde altınlarla olan mesafeler tutuluyor. A'nın konumu ile bütün altınların konumu sırasıyla $|x1-x0|+|y1-y0|$ formülü kullanılarak hesaplıyor. Bu hesaplanan değer mesafe matrisinde altının konumuna yazılıyor. For döngüleri sayesinde en kısa mesafe tespit ediliyor ve A oyuncusu hedefini belirlemiş oluyor. A oyuncusu hareket etmeden önce parasının olup olmadığı kontrol ediliyor, eğer parası yoksa öldü olarak kabul ediliyor. Hareketini ilk önce x düzleminde gerçekleştiriyor. 1 turda en fazla 3 adım ilerleyebiliyor. Bu değer menüden değiştirilebilir. A oyuncusu her adım attığında adım atma maliyeti harcıyor ve gizli altın kontrolü yapıyor. Adım atma süreci x düzleminde hedefiyle eşitlenene kadar devam ediyor. Daha sonra Y düzleminde hedefine doğru ilerliyor. Y konumunda hedefine ulaşırsa ilerlemeyi durduruyor. İlerlemeleri bittikten sonra hedefini varıp varmadığını kontrol ediyor. Eğer vardıysa hedefleri sıfırlıyor. Bulunduğu konumdaki altını alıyor ve altın sayısını güncelliyor. Bu işlemlerin hepsini

yaptıktan sonra tekrar hedef belirleme kısmı çalışıyor. Bu sayede A oyuncusu hiç hedefsiz kalmamış oluyor.

B oyuncusu ise en karlı altına doğru gitmekte. Hareket etme algoritmaları A oyuncusuyla tamamen aynı. Tek fark hedef belirleme algoritması. Hedef belirleme algoritmasında mesafe matrisi yerine kazanç matrisi kullanılıyor. $((|x1-x0|+|y1-y0|*mesafe\ maliyeti)-o\ konumdaki\ altın)+1$ formülü kullanılıyor. Bu formülde o altının karlı altın olabilmesi için en düşük sayısının çıkması lazım. “+1” eklememizin sebebi ise eğer o konumdaki kar-zarar ilişkisi 0 ise matrisin içerisindeki altınların bulunmadığı 0 konumuyla karışmaması için ekliyoruz. Bütün altınlara eklediğimiz için kar-zarar da bir değişiklik olmuyor. Eğer uzaktaki altınla yakındaki altının kar-zarar miktarı aynı ise rastgele birine gidiyor.

C oyuncusu ise B oyuncusunun aynısı tek farkı ise her tur ona geldiğinde kendisine en yakın 2(Bu sayı menüden değiştirilebilir) gizli altını açabilmesidir. Bu olay ona diğerlerinden daha önce gizli altınları görmesini sağlıyor. Açtığı gizli altınları diğer oyuncularda görebilir. Bu gizli altınları açma özelliği cSuperpower Fonksiyonu sayesinde gerçekleştirilmektedir. Bu fonksiyonda C oyuncusuna en yakın gizli altınlar tespit edilmekte. Tespit edilen en yakın gizli altınlar, CheckSecretGold adındaki gizli altın kontrolü yapan fonksiyon sayesinde altın matrisinin içerisindeki -1 değeri 5,10,15,20 değerlerinden rastgele biri seçilerek değiştirilmektedir. Bu sayede gizli altınlar normal altınlara dönüşmektedir.

D oyuncusunun çoğu özelliği B oyuncusuyla aynı şekilde çalışıyor. D oyuncusunun kendine has olan özelliği ise diğer oyuncuların hedeflerini görebilmesi. Diğer oyuncuların hedeflerini kontrol ediyor. Örnek vermek gerekirse B oyuncusu X:3 Y:3 konumunda ve hedefi X:5 Y:3 olsun. D oyuncusunun konumu ise X:1 Y:2 olsun. Bu durumda D oyuncusu hem B’nin hem de kendisinin hedefe olan uzaklığını ölçecek eğer kendisi daha uzakta ise kendi kazanç matrisinde o altının konumunu 0 olarak koyarak orayı görmezden gelmektedir. Geri kalan altınlar arasından kar-zarar ilişkisine göre ne karlı altını seçmektedir.

4-) Sonuç

Sonuçta C# dili kullanılarak, Butonlar, Fonksiyonlar, Arayüz, Dosya Sistemi, Matrisler gibi konularda program yazılarak Form yapıları düşünce tarzının ve algoritmik düşünme yetisinin gelişmesi sağlandı.

5-) Deneyisel Sonuçlar

Altın Toplama Oyunu

Oyun Tahtasının Genişliği: 20

Oyun Tahtası Yüksekliği: 20

Altın Bulunma Oranı (%): 20

Gizli Altın Bulunma Oranı (%): 10

Başlangıç Altın Miktarı: 200

Bir Turdaki Maksimum Adım Sayısı: 3

A Oyuncu Hedef Belirleme Maliyeti: 5

B Oyuncu Hedef Belirleme Maliyeti: 10

C Oyuncu Hedef Belirleme Maliyeti: 15

D Oyuncu Hedef Belirleme Maliyeti: 20

A Oyuncu Hamle Maliyeti: 5

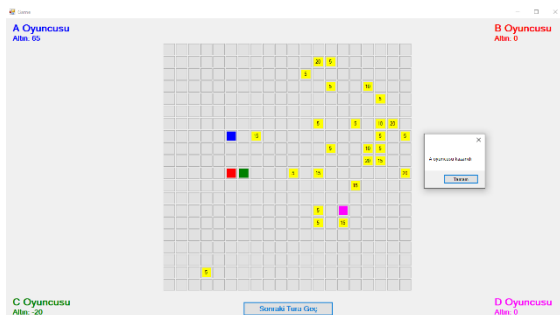
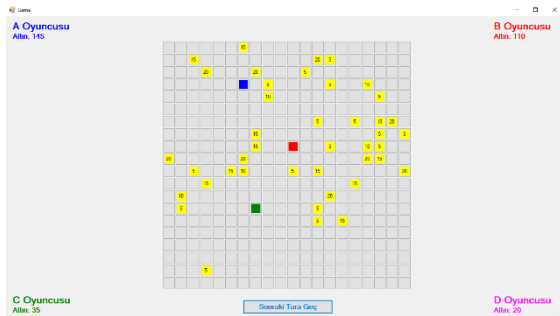
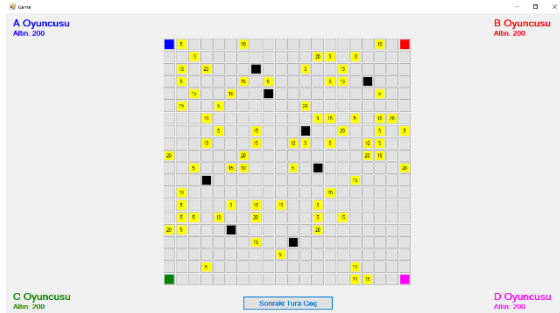
B Oyuncu Hamle Maliyeti: 5

C Oyuncu Hamle Maliyeti: 5

D Oyuncu Hamle Maliyeti: 5

C Oyuncu Gizli Altın Açma Miktarı: 2

Oyunu Başlat



Oyun Özeti

A Oyuncusu:

Toplam Adım Sayısı: 46

Harcanan Altın miktarı: 325

Kasadaki Altın miktarı: 65

Toplanan Altın miktarı: 190

B Oyuncusu:

Toplam Adım Sayısı: 46

Harcanan Altın miktarı: 430

Kasadaki Altın miktarı: 0

Toplanan Altın miktarı: 230

C Oyuncusu:

Toplam Adım Sayısı: 25

Harcanan Altın miktarı: 335

Kasadaki Altın miktarı: -20

Toplanan Altın miktarı: 115

D Oyuncusu:

Toplam Adım Sayısı: 27

Harcanan Altın miktarı: 315

Kasadaki Altın miktarı: 0

Toplanan Altın miktarı: 115

X:0 Y:0

X:1 Y:0

X:1 Y:1

X:1 Y:2

X:1 Y:3

X:1 Y:4

X:1 Y:5

X:2 Y:5

X:2 Y:4

X:3 Y:4

X:3 Y:5

X:3 Y:6

X:3 Y:7

X:3 Y:8

X:4 Y:8

X:4 Y:7

X:4 Y:6

X:4 Y:5

X:5 Y:5

X:5 Y:4

X:6 Y:4

X:6 Y:3

X:7 Y:3

X:8 Y:3

X:8 Y:4

X:7 Y:4

X:7 Y:3

X:7 Y:2

X:6 Y:2

X:6 Y:1

X:6 Y:0

X:5 Y:0

X:4 Y:0

X:3 Y:0

X:3 Y:1

X:3 Y:2

X:2 Y:2

X:2 Y:1

X:2 Y:2

X:2 Y:3

X:2 Y:4

X:2 Y:5

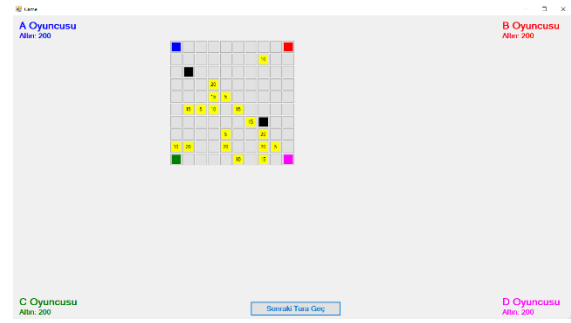
X:2 Y:6

X:2 Y:7

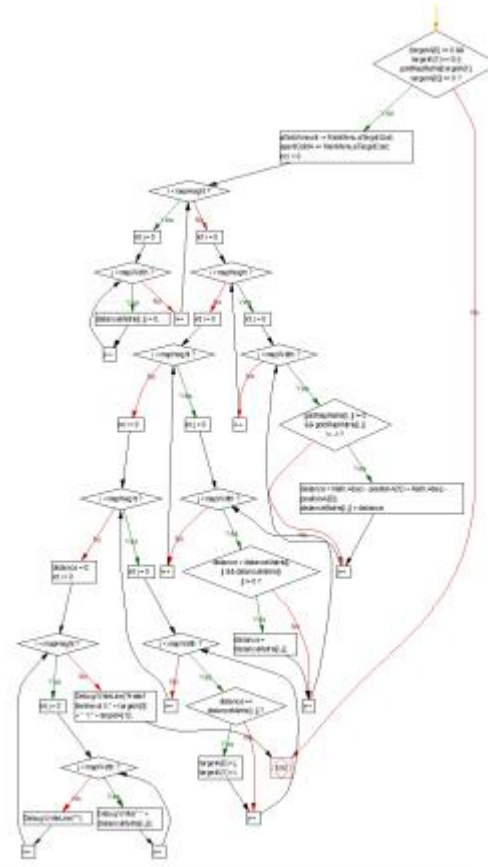
X:3 Y:7

X:4 Y:7

X:5 Y:7



6-) Akış Diyagramı



7-) Kaynakça

- https://www.youtube.com/watch?v=_ltxEwumJ9FY&list=WL&index=35
- <https://www.youtube.com/watch?v=vC7ltdj7rYY&list=WL&index=34>
- <https://www.geeksforgeeks.org/csharp-program-for-dijkstras-shortest-path-algorithm-greedy-algo-7/>
- <https://www.tutlane.com/tutorial/csharp/csharp-string-join-method>