BİLGİSAYAR BİLİMLERİNE GİRİŞ DÖNEM PROJESİ

Ahmet Salih Meydan 21011047

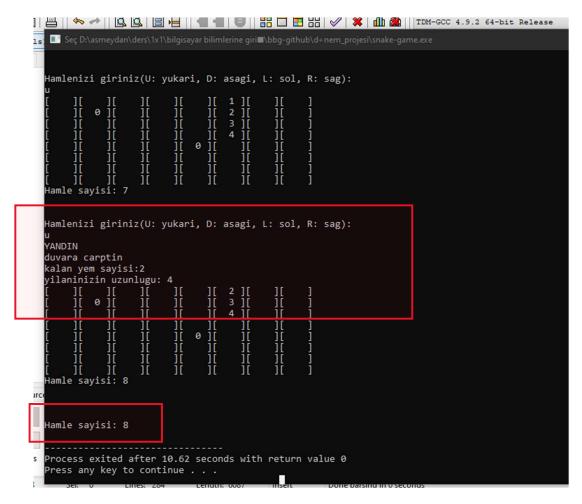
Oyunu çalıştırınca öncelikle oyun alanını oluşturmak için bir satır bir de sütun sayısı isteniyor ardından oyun alanında kaç tane yemek bulunması gerektiğini soruyor. Bu değerler girildikten sonra rastgele bir yılan yine kullanıcıdan alınan sayı kadar rastgele konumda yemeğin bulunduğu bir haritayı yazdırıyor. Ardından yılanı yönlendirmemiz için hamle girmemiz gerektiğini belirtiyor.

Girilen hamle u olduğu için yılan 1 birim yukarı gitmiştir (u harfinin küçük veya büyük olması önemli değildir ikisi de kabul edilir). Ardından yeni hamle beklenmektedir. Her hamlede hamle sayısı yazmaktadır

2 kere l hamlesi oynandığı için yılan yemi yiyor fakat henüz büyümüyor çünkü bir sonraki hamlede büyüyecek

Resimde de görüldüğü üzere yemeğin yenilişinden sonraki hamlede yılan büyümüş ve 1-2 halini almıştır.

Bütün yemekler yenildikten sonra "Tebrikler kazandınız "yazısı ve hamle sayısı ekrana yazdırılıp oyun biter.



Eğer yılan duvara çarparsa YANDIN, duvara çarptın, kalan yem sayısı, yılanın uzunluğu ve hamle sayısı yazdırılır. Ardından oyun biter.

•

Eğer yılan kendi kuyruğuna basarsa ekranda YANDIN, kuyruğa bastın, kalan yem sayısı, yılanın uzunluğu ve hamle sayısı yazdırılır. Ardından oyun biter.

Yemeklerin ve yılanın başlangıçtaki yerini belirlemek ve yemeklerin üst üste gelmesini engellemek için N * M boyutunda bir dizi oluşturdum.

Yemek sayısı N * M den büyük olamayacağından eğer N * M den büyük girilirse tekrar istenecek. Ardından rand fonfsiyonuyla oluşturulan rastgele bir sayı random dizisindeki rastgele sayının indeksindeki sayının ifade ettiği bölgeye yemek yerleştirilir ve bu sayı diziden çıkarılır. İstenilen sayı kadar yemek yerleştirildikten sonra aynı işlem yılanın kafası yani 1 sayısı için yapılır.

```
62

63

64

// HARİTAYI YAZDIRMA

for(i = 0; i < N; i++) {

for(j = 0; j < M; j++) {

if(!(map[i][j] == -1)) {

printf("[ %d ]", map[i][j]);

}

else {

printf("[ ]", map[i][j]);

}

printf("[ ]", map[i][j]);

}

printf("\n");

}
```

Haritanın matrisinde yılan 1den büyük sayılarda yemekler 0 olarak boşluklar da -1 olarak tutulduğu için haritayı yazdırırken -1ler boşluk diğer sayılar kendi sayısını yazdırmaktadır.

Bir hamle yapılınca öncelikle haritadaki 0 dan büyük bütün sayılar bir arttırılır ardından oynanılan hamleye göre gidilen yönde duvar veya kuyruğun bir parçası olup olmadığına bakılır eğer bir sıkıntı yoksa haritadaki yılanın uzunluğundan büyük sayıları -1 yani boşluk yapılır. Gidilen bölgede yem yenilip

yenilmediğine de bakıldıktan sonra da yılanın kafası yani 1 sayısı koyulup yılanın kafasının tutulduğu değer değiştirilir.

```
267
268
269
270
271
272
272
273
274
275
276
277
28
278
279
280
281
282
283
284

Resources Compile Log Debug Find Results Close

//YEM YENMİŞSE
if (yem) {
    y_uzunluk += 1;
    puan += 1;
    if (puan == yemek_sayisi) {
        flag = 1;
        printf("\n Tebrikler kazandiniz\n");
        }
        yem = 0;
}

Resources Compile Log Debug Find Results Close
```

Hamle yapıldıktan sonra yemek yenilip yenilmediği kontrol ediliyordu eğer yemek yenmişse yılanın uzunluğu 1 arttırılır puan 1 arttırılır eğer haritadaki yemek sayısı puana eşitse kazandınız mesajı verilir ve en son da tekrar yemek yenilip yenilmediğini kontrol edebilmek için yem değişkeni sıfır yapılır.

Video linki:

https://www.youtube.com/watch?v=9yPL5CtNgEo