

BİLGİSAYAR BİLİMLERİNE GİRİŞ DÖNEM PROJESİ

Ahmet Salih Meydan

21011047

```
D:\asmeydan\ders\1x1\bilgisayar bilimlerine giri\bbg-github\d+nem_projesi\snake-game.exe
Oyun alanini giriniz(N x M):
7
7
N: 7 M: 7
Yemek sayisini giriniz(oyun alanindan buyuk olamaz):
5
[ ][ 0 ][ ][ ][ ][ 0 ][ ][ ]
[ ][ ][ ][ ][ ][ ][ ][ ]
[ ][ ][ ][ ][ ][ 0 ][ ][ ]
[ ][ ][ ][ 1 ][ ][ ][ ][ ]
[ ][ ][ ][ ][ ][ ][ ][ ]
[ ][ ][ ][ ][ 0 ][ ][ ][ ]
[ ][ ][ ][ ][ ][ ][ ][ 0 ]
Hamlenizi giriniz(U: yukari, D: asagi, L: sol, R: sag):
```

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.


```
D:\asmeydan\ders\1x1\bilgisayar bilimlerine giri\bbg-github\d-nem_projesi\snake-game.exe

Hamlenizi giriniz(U: yukari, D: asagi, L: sol, R: sag):
d
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
[  ][  ][  ][  ][  ][  ][ 2 ][ 3 ][ 4 ][  ]
[  ][  ][  ][  ][  ][  ][ 1 ][  ][ 5 ][  ]
[  ][  ][  ][  ][  ][  ][ 0 ][  ][ 6 ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][ 7 ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
Hamle sayisi: 29

Hamlenizi giriniz(U: yukari, D: asagi, L: sol, R: sag):
d
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
[  ][  ][  ][  ][  ][  ][ 3 ][ 4 ][ 5 ][  ]
[  ][  ][  ][  ][  ][  ][ 2 ][  ][ 6 ][  ]
[  ][  ][  ][  ][  ][  ][ 1 ][  ][ 7 ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
[  ][  ][  ][  ][  ][  ][  ][  ][  ][  ]
Hamle sayisi: 30

Tebrikler kazandiniz
Hamle sayisi: 30

-----
Process exited after 760.1 seconds with return value 0
Press any key to continue . . .
```

Bütün yemekler yenildikten sonra “Tebrikler kazandiniz” yazısı ve hamle sayısı ekrana yazdırılıp oyun biter.


```

20 }
21
22
23 // RASTGELELİK İÇİN DİZİ OLUŞTURMA
24 int random[N * M];
25 for(i = 0; i < N; i++) {
26     for(j = 0; j < M; j++) {
27         random[k] = k;
28         k += 1;
29     }
30 }
31
32

```

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

```

32
33 // YEMEK SAYISINI ALMA
34 do {
35     printf("Yemek sayisini giriniz(oyun alanından buyuk olamaz): \n");
36     scanf("%d", &yemek_sayisi);
37 }
38 while (yemek_sayisi >= N * M);
39
40 // YEMEKLERİN YERLERİNİ BELİRLEME
41 // 2 YEMEĞİN ÜST ÜSTE OLMAMASINI SAĞLAMA
42 for(i = 0; i < yemek_sayisi; i++) {
43     sayi = rand()%k;
44     map[random[sayi] / M][random[sayi] % M] = 0;
45     for(j = 0; j < k; j++) {
46         if(random[j] == random[sayi]) {
47             for(l = j; l < (k - 1); l++) {
48                 random[l] = random[l + 1];
49             }
50         }
51     }
52     random[k - 1] = '\0';
53     k -= 1;
54 }
55
56 // YILAN
57 sayi = rand()%k;
58 map[random[sayi] / M][random[sayi] % M] = 1;
59 y_uzunluk = 1;
60 yN = random[sayi] / M;
61 yM = random[sayi] % M;
62
63

```

Yemek sayısı $N * M$ den büyük olamayacağından eğer $N * M$ den büyük girilirse tekrar istenecek. Ardından rand fonksiyonuyla 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
65 for(i = 0; i < N; i++) {
66     for(j = 0; j < N; j++) {
67         if(!(map[i][j] == -1)) {
68             printf("[ %d ]", map[i][j]);
69         }
70         else {
71             printf("[   ]", map[i][j]);
72         }
73     }
74     printf("\n");
75 }
76

```

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

```

169 // R'YA BASILINCA
170 if((hamle == 'R') || (hamle == 'r')) {
171     for(i = 0; i < N; i++) {
172         for(j = 0; j < N; j++) {
173             if(map[i][j] > 0) {
174                 map[i][j] += 1;
175             }
176         }
177     }
178     if((yM + 1 > M - 1) || (map[yN][yM + 1] > 0)) {
179         flag = 1;
180         printf(" YANDIN\n");
181         if(yM + 1 > M - 1) {
182             printf("duvara carptin\n");
183         }
184         else if(map[yN][yM + 1] > 0) {
185             printf("kuyruğa bastın\n");
186         }
187         kalan = yemek_sayisi - puan;
188         printf("kalan yem sayisi:%d \n", kalan);
189         printf("yılanınızın uzunluğu: %d \n", y_uzunluk);
190     }
191     for(i = 0; i < N; i++) {
192         for(j = 0; j < N; j++) {
193             if(map[i][j] > y_uzunluk) {
194                 map[i][j] = -1;
195             }
196         }
197     }
198     if(map[yN][yM + 1] == 0) {
199         yem = 1;
200     }
201     map[yN][yM + 1] = 1;
202     yM += 1;
203 }
204 // L'YA BASILINCA
205

```

Bir hamle yapılıncaya 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 //YEM YENMİŞSE
270 if(yem) {
271     y_uzunluk += 1;
272     puan += 1;
273     if(puan == yemek_sayisi) {
274         flag = 1;
275         printf("\n Tebrikler kazandınız\n");
276     }
277 }
278 yem = 0;
279 }
280
281 printf("\nHamle sayisi: %d\n", hamle_sayisi);
282
283 return 0;
284 }
```

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>