

2020-2021 Güz Yarıyılı
Algoritma Analizi
Ödev – 1

Ders Yürütücüleri
Doç. Dr. M. Elif KARSLIGİL
Dr. Öğr. Üyesi M. Amaç GÜVENSAN

Burak Boz

18011706

Soru 2

Problem: Bir matriste, verilen N değeri dikkate alınarak, “Von Neumann’s Neighborhood” kuralına göre (<https://mathworld.wolfram.com/vonNeumannNeighborhood.html>) hücrelerin “1” değeri ile işaretlenmesi isteniyor. Aşağıda N=0,1,2 değerlerine göre matris içerisindeki hücreler siyah ile gösterilmektedir (Beyaz=“0”, Siyah = “1”).

Von Neumann’s Neighborhood” kuralını gözeterek verilen N değerine göre matrisleri oluşturan programın kodunu C dilinde yazınız. Program çıktısı, ilgili matrisi ve satır başına ve toplam siyah hücre sayısını göstermelidir. Matris dinamik bellek ayırma yöntemi ile en uçtaki siyah hücrelerin dışında sadece bir satır ve sütun kalacak şekilde oluşturulmalıdır. Örnek N=0 için 3x3, N=1 için 5x5 boyutunda olmalıdır. Şekildeki matris boyutlarını DİKKATE ALMAYINIZ. (60 Puan)

Çözüm: Kullanıcıdan n değeri alınır. Bu alınan n değerine göre matrisin boyutu hesaplanır. Bu boyut $((n+1)*2)+1$ şeklinde hesaplanır. Matrisin orta noktası oluşturulan matrise göre hesaplanır. Daha sonra döngüler ile gerekli değerler matris içerisine yazdırılır. Bu değerler yazdırılırken satırlardaki yazılması gereken 1 değeri ve her satırda kaç adet 1 olacağı, orta noktanın bulunduğu satıra gelene kadar arttırılır. Daha sonra azaltılır. Oluşturulan matris ve döngüler, köşe satırları sıfır bırakacak şekilde tasarlanmıştır. Merkez noktadan tüm 1 ile işaretlenmiş noktalara maksimum n adımda gidilebilir. Programın son kısmında her satırda kaç adet 1 olduğu ve toplamda kaç adet 1 kullanıldığı yazdırılmaktadır.

Karmaşıklık: $(2*n*(n+1)+1)*n$
 $=n^3$

Ekran Çıktıları:

Ekran çıktısı 1:

Girdi: n sayısı =1

Çıktı: Kullanılan 1 sayısı : 5

```
C:\Users\user\Desktop\oku\3-1\Algoritma Analizi\dev1\Soru2\vonN.exe
*** Von Neumann Neighborhood | Soru 2 - Burak Boz ***

Lutfen n sayisini girin:1
2. satirda 1 adet 1 bulunmaktadır
3. satirda 3 adet 1 bulunmaktadır
4. satirda 1 adet 1 bulunmaktadır

00000
00100
01110
00100
00000

Kullanilan 1 sayisi:5
-----
Process exited after 0.7151 seconds with return value 23
Press any key to continue . . .
```

Ekran çıktısı 2:

Girdi: n sayısı =3

Çıktı: Kullanılan 1 sayısı : 25

```
C:\Users\user\Desktop\okul3-1\Algoritma Analizi\dev1\Soru2\vonN.exe
*** Von Neumann Neighborhood | Soru 2 - Burak Boz ***

Lutfen n sayisini girin:3
2. satirda 1 adet 1 bulunmaktadir
3. satirda 3 adet 1 bulunmaktadir
4. satirda 5 adet 1 bulunmaktadir
5. satirda 7 adet 1 bulunmaktadir
6. satirda 5 adet 1 bulunmaktadir
7. satirda 3 adet 1 bulunmaktadir
8. satirda 1 adet 1 bulunmaktadir

000000000
000010000
000111000
001111100
011111110
001111100
000111000
000010000
000000000

Kullanilan 1 sayisi:25
-----
Process exited after 1.695 seconds with return value 24
Press any key to continue . . .
```

Ekran çıktısı 3:

Girdi: n sayısı =5

Çıktı: Kullanılan 1 sayısı : 61

```
C:\Users\user\Desktop\okul3-1\Algoritma Analizi\dev1\Soru2\vonN.exe
*** Von Neumann Neighborhood | Soru 2 - Burak Boz ***

Lutfen n sayisini girin:5
2. satirda 1 adet 1 bulunmaktadir
3. satirda 3 adet 1 bulunmaktadir
4. satirda 5 adet 1 bulunmaktadir
5. satirda 7 adet 1 bulunmaktadir
6. satirda 9 adet 1 bulunmaktadir
7. satirda 11 adet 1 bulunmaktadir
8. satirda 9 adet 1 bulunmaktadir
9. satirda 7 adet 1 bulunmaktadir
10. satirda 5 adet 1 bulunmaktadir
11. satirda 3 adet 1 bulunmaktadir
12. satirda 1 adet 1 bulunmaktadir

0000000000000
0000001000000
0000011100000
0000111110000
0001111111000
0011111111100
0111111111110
0011111111100
0001111110000
0000111100000
0000011100000
0000001000000
0000000000000

Kullanilan 1 sayisi:61
-----
Process exited after 1.257 seconds with return value 24
Press any key to continue . . .
```

Ekran çıktısı 4:

Girdi: n sayısı =6

Çıktı: Kullanılan 1 sayısı : 85

```
C:\Users\user\Desktop\okul\3-1\Algoritma Analizi\dev1\Soru2\vonN.exe
*** Von Neumann Neighborhood | Soru 2 - Burak Boz ***

Lutfen n sayisini girin:6
2. satirda 1 adet 1 bulunmaktadır
3. satirda 3 adet 1 bulunmaktadır
4. satirda 5 adet 1 bulunmaktadır
5. satirda 7 adet 1 bulunmaktadır
6. satirda 9 adet 1 bulunmaktadır
7. satirda 11 adet 1 bulunmaktadır
8. satirda 13 adet 1 bulunmaktadır
9. satirda 11 adet 1 bulunmaktadır
10. satirda 9 adet 1 bulunmaktadır
11. satirda 7 adet 1 bulunmaktadır
12. satirda 5 adet 1 bulunmaktadır
13. satirda 3 adet 1 bulunmaktadır
14. satirda 1 adet 1 bulunmaktadır

0000000000000000
0000000100000000
0000001110000000
0000011111000000
0000111111100000
0001111111110000
0011111111111000
0011111111111100
0111111111111110
0011111111111100
0001111111111000
0000111111111000
0000011111100000
0000001111100000
0000000110000000
0000000010000000
0000000000000000

Kullanilan 1 sayisi:85
-----
Process exited after 0.9539 seconds with return value 24
Press any key to continue . . .
```

Kodlar ve Tanımlar:

Kullanılan değişkenler:

int i=0;//Döngülerde kullanılacak değişken

int j=0;//Döngülerde kullanılacak değişken

int **matris;//Matrisin oluşturulması

int n=0;//Komşu sayısını belirlemek için kullanılan ana değişken

int matrisBoyut=((n+1)*2)+1;//Alınan ana değişkene göre matris boyutunun hesaplanması

int ortaNokta=(matrisBoyut/2);//Merkez nokta

int adet=1;//Her satırda kaç adet 1 olacağını belirlemek için kullanılan değişken

int baslangicNoktasi=ortaNokta;//Her satırda 1 lemeye başlarken kullanılacak başlangıç noktası

int tmp=0;//Başlangıç noktasının değerinin kaybolmaması için tutulan geçici değişken

int hucreSayisi=0;//Kullanılan 1 sayısının hesaplanmasında kullanılacak değişken

Kod:

/*

07.11.2020

Burak Boz

18011706

Von Neumann Neighborhood

*/

#include <stdio.h>

void main()

{

printf("*** Von Neumann Neighborhood | Soru 2 - Burak Boz ***\n\n");

int i=0;//Döngülerde kullanılacak değişken

int j=0;//Döngülerde kullanılacak değişken

int **matris;//Matrisin oluşturulması

printf("Lutfen n sayisini girin:");

int n=0;//Komşu sayısını belirlemek için kullanılan ana değişken

scanf("%d",&n);

int matrisBoyut=((n+1)*2)+1;//Alınan ana değişkene göre matris boyutunun hesaplanması

int ortaNokta=(matrisBoyut/2);//Merkez nokta

matris = (int **)calloc(matrisBoyut,sizeof(int*));

for(i=0;i<matrisBoyut;i++)//Bu döngüde matris sıfırlanıyor

{

matris[i] = (int*)calloc(matrisBoyut,sizeof(int));

}

int adet=1;//Her satırda kaç adet 1 olacağını belirlemek için kullanılan değişken

int baslangicNoktasi=ortaNokta;//Her satırda 1 lemeye başlarken kullanılacak başlangıç noktası

int tmp=0;//Başlangıç noktasının değerinin kaybolmaması için tutulan geçici değişken

for(i=1;i<matrisBoyut-1;i++)//Bu döngüde en üst ve en alt satır hariç tüm satırlar işlenecek

{

if(i<=ortaNokta)//Matrisin üst bölgesini kontrol etmek için

```

        //Bu döngüde üst bölgedeki satırlara 1'ler yerleştiriliyor

        printf("%d. satirda %d adet 1 bulunmaktadır\n",i+1,adet);

        tmp=baslangicNoktasi;
        for(j=0;j<adet;j++)
        {
            matris[i][baslangicNoktasi]=1;

            baslangicNoktasi++;

        }

        baslangicNoktasi=tmp;
        adet=adet+2;
        baslangicNoktasi--;
    }

    else//Matrisin alt bölgesini kontrol etmek için
    //Bu döngüde alt bölgedeki satırlara 1'ler yerleştiriliyor

        if(i==ortaNokta+1)//Bölgeler arasındaki geçiş için gerekli kontrol

        {

            adet=adet-2;

            baslangicNoktasi++;

        }

        adet=adet-2;
        baslangicNoktasi++;
        tmp=baslangicNoktasi;
        for(j=0;j<adet;j++)
        {

            matris[i][baslangicNoktasi]=1;

            baslangicNoktasi++;

        }

        baslangicNoktasi=tmp;

        printf("%d. satirda %d adet 1 bulunmaktadır\n",i+1,adet);

    }

}

printf("\n\n");

//Matrisin yazdırılması için gerekli kodlar

```

```
for(i=0;i<matrisBoyut;i++)
{
    for(j=0;j<matrisBoyut;j++)
    {
        printf("%d",matris[i][j]);

    }
    printf("\n");
}

int hucreSayisi=0;//Kullanılan 1 sayısının hesaplanmasında kullanılacak değişken

hucreSayisi = 2*n*(n+1)+1;

printf("\n\nKullanılan 1 sayısı:%d",hucreSayisi);

}
```