



BLM3021

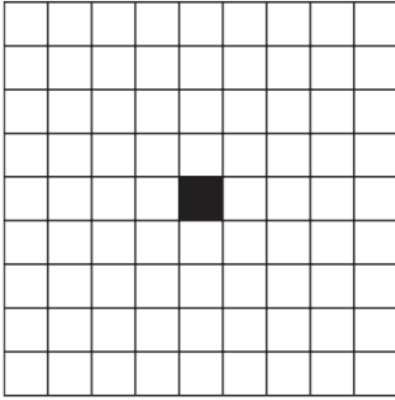
ALGORİTMA ANALİZİ

ÖDEV-1

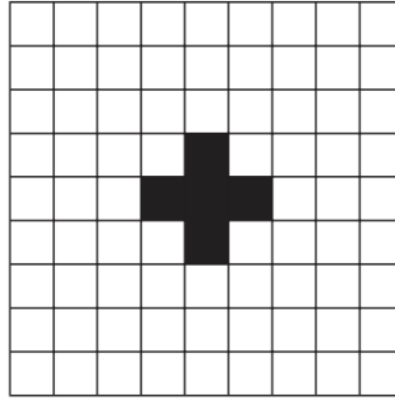
SORU 2

Problem 2: 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”).

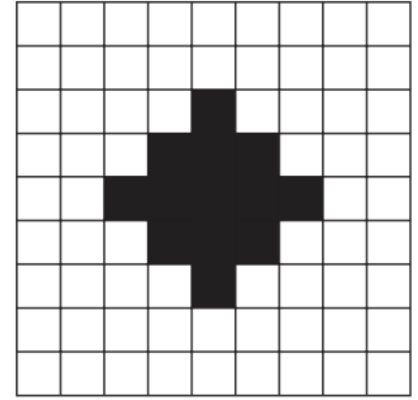
Von Neumann’s Neighborhood” kuralını gözeterek verilen N değerine göre matrisleri oluşturan programın kodunu C dilinde yazınız. Program çıktısı, ilgili matrisi ve satır başına ve toplam siyah hücre sayısını göstermelidir. Matris dinamik bellek ayırma yöntemi ile en uçtaki siyah hücrelerin dışında sadece 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)



$n = 0$



$n = 1$



$n = 2$

ÇÖZÜM

Kodlar:

```
#include <stdio.h>
#include <stdlib.h>

int bCount[250]={0}; //bu dizi ile elde edilen siyah adetlerin saym yapılır. Initiate değeri olarak 0 verilmiştir.

void createArrayWithZeros(int **array,int arrayLenght){
    int i,j;
    for(i=0;i<arrayLenght;i++){
        for(j=0;j<arrayLenght;j++){ //matrisimizin tüm elemanların 0 ile dolduruyoruz.
            array[i][j]=0;
        }
    }
}

void fillArrayWithOnes(int **array,int arrayLenght,int N){
    int i,j;
    for(i=0;i<arrayLenght;i++){
        for(j=0;j<arrayLenght;j++){
            if(abs(i-N)+abs(j-N)<=N){ //gerekli kontrol yapıyor. Ortta varsa ise;
                array[i][j]=1; //ilgili hücrenin değeri 1leniyor.
                bCount[i]++; //o satıra ait sayacın değeri 1 arttırılıyor.
            }
        }
    }
}

}
```

```

void showArray(int **array,int arrayLenght,int N){
    int i,j;
    for(i=0;i<arrayLenght;i++){//basit eklede arrayimizi print ettik
        for(j=0;j<arrayLenght;j++){
            printf("%d ",array[i][j]);
        }
        printf("\n");
    }
    printf("\n");
    for(i=0;i<arrayLenght;i++){//satr saysi kadar,satrlarda ka
        printf("Satir NO=%d <-> Siyah kare adeti=%d \n",i,bCount[i]);
    }

    printf("Toplam siyah kare adeti=%d",(2*N*(N+1))+1);//toplam siyah kare adetinide
    printf("\n");
}

int main(int argc, char *argv[]) {
    int N,arrayLenght,**array;//gerekli degisken tanmlamalar.

    printf("N degerini girin:");//N degeri kullancladan istenilir ve degiskene atilir.
    scanf("%d",&N);

    arrayLenght=(2*N)+3;//verilen N degiskenine g
    printf("\n");
    //dizinin boyutu belirlenir.
    //az alrsak.N=0 i
    printf("\n");
    array=(int**)calloc(arrayLenght,sizeof(int*));//dizi bellekte alan
    printf("\n");
    istenmesi.(4*diziBoyutu) Byte
    int i,j;
    for(i=0;i<arrayLenght;i++){
        array[i]=(int*)calloc(arrayLenght,sizeof(int));
    }

    createArrayWithZeros(array,arrayLenght);//0lar ile doldurarak matrisi oluturacagmz
    printf("\n");
    fillArrayWithOnes(array,arrayLenght,N);//gerekli yerleri tespit edip 1 de
    printf("\n");
    showArray(array,arrayLenght,N);//arrayimizi g
    printf("\n");
    free(array);//btn ilemler sonunda arraymz bellekten freeliyoruz.

    return 0;
}

```

Ekran çıktıları:

```
N degerini girin:0
0      0      0
0      1      0
0      0      0

Satir NO=0      <->      Siyah kare adeti=0
Satir NO=1      <->      Siyah kare adeti=1
Satir NO=2      <->      Siyah kare adeti=0
Toplam siyah kare adeti=1
-----
Process exited after 8.807 seconds with return value 0
Press any key to continue . . .
```

```
N degerini girin:2
0      0      0      0      0      0      0
0      0      0      1      0      0      0
0      0      1      1      1      0      0
0      1      1      1      1      1      0
0      0      1      1      1      0      0
0      0      0      1      0      0      0
0      0      0      0      0      0      0

Satir NO=0      <->      Siyah kare adeti=0
Satir NO=1      <->      Siyah kare adeti=1
Satir NO=2      <->      Siyah kare adeti=3
Satir NO=3      <->      Siyah kare adeti=5
Satir NO=4      <->      Siyah kare adeti=3
Satir NO=5      <->      Siyah kare adeti=1
Satir NO=6      <->      Siyah kare adeti=0
Toplam siyah kare adeti=13
-----
Process exited after 6.072 seconds with return value 0
Press any key to continue . . .
```

N degerini girin:6

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Satir NO=0 <-> Siyah kare adeti=0
Satir NO=1 <-> Siyah kare adeti=1
Satir NO=2 <-> Siyah kare adeti=3
Satir NO=3 <-> Siyah kare adeti=5
Satir NO=4 <-> Siyah kare adeti=7
Satir NO=5 <-> Siyah kare adeti=9
Satir NO=6 <-> Siyah kare adeti=11
Satir NO=7 <-> Siyah kare adeti=13
Satir NO=8 <-> Siyah kare adeti=11
Satir NO=9 <-> Siyah kare adeti=9
Satir NO=10 <-> Siyah kare adeti=7
Satir NO=11 <-> Siyah kare adeti=5
Satir NO=12 <-> Siyah kare adeti=3
Satir NO=13 <-> Siyah kare adeti=1
Satir NO=14 <-> Siyah kare adeti=0

Toplam siyah kare adeti=85

Set: 0 Lines: 60 Length: 2427 Insert Done parsing in 0.010 seconds