



# **VERİ YAPILARI VE ALGORİTMALAR PROJE**

**GRAF ÜZERİNDE ENİNE ARAMA(BREADTH  
FIRST SEARCH-BFS)KULLANIMI İLE KEVIN  
BACON SAYISININ(KÜÇÜK DÜNYA  
HİPOTEZİ) BULUNMASI**

## PROJENİN ADIM ADIM AÇIKLANMASI VE YÜRÜTÜLMESİ:

### 1)Proje konusunun açıklanması ve projenin ana yürütölme mantığı:

Projede Küçük Dünya Hipotezinden (Small World Phenomenon) yararlanılarak hazır verilen, içerisinde film isimleri ve bu filmlerde oynayan oyuncuların bulunduğu dosyaları okuyarak BFS(Breadth First Search yöntemi) ile dışardan kullanıcının girdiği bir film oyuncusunun diğer bir film oyuncusu olan **Kevin Bacon**'a olan uzaklığını hesaplayan program tasarlanmıştır.Bunu yaparken oyuncu olan **Kevin Bacon**'ı baz alarak uzaklık hesabı yapılmaktadır.Kullanıcı tarafından girilen oyuncu **Kevin Bacon** ile aynı filmde oynamışsa,bu oyuncunun **Kevin Bacon** sayısı 1 olur. **Kevin Bacon** girildiği taktirde bu sayı 0 olur.Farklı filimler aracılığı ile oyuncuların birbiriyle olan zincirinin sonucunda kullanıcı tarafından girilen oyuncu zincirin sonunda **Kevin Bacon** ile bağı çıkar ise aradaki zincir sayısı kadar **Kevin Bacon** sayısına sahiptir.Hiç bağı yok ise bu sayı sonsuzdur.

2)Programın akışında ise;Başlangıçta kullanıcıdan .txt formatında dosya ismi alınır daha sonra 2 seçenek sunulur,harici verilen seçeneklerde program kapatılır.Bu seçenekler;

1)Aktörlerin ve filmlerin okunmasından sonra tüm aktörlerin total Kevin Bacon sayılarını satır satır ekrana yazdırır.

2)Spesifik olarak verilen oyuncu adına göre oyuncunun Kevin Bacon sayısı bulunur.

### 3)Projede kullanılan BFS algoritmasının açıklaması,akışı:

3.1)Programda BFS algoritmasını kullanan 2 fonksiyon vardır bunlardan birtanesi dosyadan okunan tüm aktörlerin Kevin Bacon sayısının bulunması için kullanılır diğeri ise spesifik olarak parametreleri yollanan aktörün verileriyle sadece o aktörün Kevin Bacon sayısını bulmaktadır.(Program içerisindeki main fonksiyonunda, BFS fonksiyonlarının kullanıldığı yerler komut satırı ile (1) ve (2) olarak gösterilmektedir) İki fonksiyon arasındaki ayrımlar;

Özel olarak girilen aktörün Kevin Bacon sayısını bulan fonksiyonda( **int BFS(struct queue\*\* kuyruk, int startingIndex, int index, int type)** )başta index kontrolü yapılır sebebi ise klavyeden girilen aktör Kevin Baconmu?Öyleyse 0 gönderir ve Kevin Bacon sayısı 0dır.

Diğer okunan tüm aktörlerin Kevin Bacon sayısını bulan fonksiyonda ise( **void BFSforAll(struct queue\*\* kuyruk, int startingIndex)** ) frekansları tutan bir dizi bulunur ve bu dizinin indisleri içerisinde bulan değerlere göre arttırım yapılarak Kevin Bacon sayısı bulunur ve ekrana yazdırılır.Örnek;

xx kişinin Kevin Bacon sayısı=1

yy kişinin Kevin Bacon sayısı=2

....

....

şeklinde.

3.2)Algoritmanın ana akış mantığı ise;

- 1)Queue boş ise fonksiyondan çık. -> STOP
- 2)Değil ise içeri gir,queue'nun en üstündeki node'u al.
- 3)Alınan node'un bütün childlarını döngü ile gez.
- 4)Önceden hiç gezilmeyen childlarını queueya ekle.
- 5)Mevcut node üzerindeki gezinti işlemi bitti,queuedan sil.
- 6)Tekrardan 1. Adıma dön.

**NOT:**Program içerisindeki yorum satırlarında gerekli işlem yerlerinde açıklamalar yapılmıştır.Yukarıda sadece kullanılan algoritmanın ana işlem adımları anlatılmıştır ve bu şekildedir.

## EKRAN GÖRÜNTÜLERİ:

### Örnek 1.

#### 1.1)Input-1.txtdeki Kevin Bacon Sayıları:

```
C:\Users\Onur Koçkan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-1.txt
input-1.txt Adli Dosyanın Acilma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:1
1 Kisinin Kevin Bacon Sayisi = 0.
1494 Kisinin Kevin Bacon Sayisi = 1.
3 Kisinin Kevin Bacon Sayisi = Sonsuz
-----
Process exited after 3.467 seconds with return value 0
Press any key to continue . . .
```

#### 1.2)Input-2.txtdeki Kevin Bacon Sayıları:

```
C:\Users\Onur Koçkan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-2.txt
input-2.txt Adli Dosyanın Acilma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:1
1 Kisinin Kevin Bacon Sayisi = 0.
61 Kisinin Kevin Bacon Sayisi = 1.
25 Kisinin Kevin Bacon Sayisi = 2.
36 Kisinin Kevin Bacon Sayisi = 3.
120 Kisinin Kevin Bacon Sayisi = 4.
182 Kisinin Kevin Bacon Sayisi = 5.
248 Kisinin Kevin Bacon Sayisi = 6.
76 Kisinin Kevin Bacon Sayisi = 7.
302 Kisinin Kevin Bacon Sayisi = 8.
237 Kisinin Kevin Bacon Sayisi = 9.
43 Kisinin Kevin Bacon Sayisi = 10.
6 Kisinin Kevin Bacon Sayisi = 11.
1176 Kisinin Kevin Bacon Sayisi = Sonsuz
-----
Process exited after 4.422 seconds with return value 0
Press any key to continue . . .
```

### 1.3)Input-3.txtdeki Kevin Bacon Sayıları:

```
C:\Users\Onur Koçkan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-3.txt
input-3.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GIRISLERINI YAPIN:1
1 Kisinin Kevin Bacon Sayisi = 0.
1372 Kisinin Kevin Bacon Sayisi = 1.
93798 Kisinin Kevin Bacon Sayisi = 2.
72980 Kisinin Kevin Bacon Sayisi = 3.
1636 Kisinin Kevin Bacon Sayisi = 4.
14 Kisinin Kevin Bacon Sayisi = 5.
717 Kisinin Kevin Bacon Sayisi = Sonsuz
-----
Process exited after 260 seconds with return value 0
Press any key to continue . . .
```

## Örnek 2.

### 2.1)Meryl Streep:

```
C:\Users\Onur Koçkan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-3.txt
input-3.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GIRISLERINI YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Streep, Meryl
River Wild, The (1994)
Bacon, Kevin
Streep, Meryl'nin Kevin Bacon Sayisi=1.

-----
Process exited after 1071 seconds with return value 0
Press any key to continue . . .
```

## 2.2) Nicolas Cage:

```
C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-3.txt
input-3.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLİYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Cage, Nicolas
Rumble Fish (1983)
Dillon, Matt
Wild Things (1998)
Bacon, Kevin
Cage, Nicolas nin Kevin Bacon Sayısı=2.

-----
Process exited after 707.5 seconds with return value 0
Press any key to continue . . .
```

## 2.3) Elie Samaha:

```
C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-3.txt
input-3.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLİYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Samaha, Elie
20 Dates (1998)
Carrere, Tia
Hostile Intentions (1994)
Pickens Jr., James
Sleepers (1996)
Bacon, Kevin
Samaha, Elie nin Kevin Bacon Sayısı=3.

-----
Process exited after 294 seconds with return value 0
Press any key to continue . . .
```

## 2.4)Dakota Fanning:

```
C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-3.txt
input-3.txt Adli Dosyanın Acilma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GIRISLERINI YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Fanning, Dakota
I Am Sam (2001)
Dern, Laura
Novocaine (2001)
Bacon, Kevin
Fanning, Dakota'nın Kevin Bacon Sayısı=2.

-----
Process exited after 257.4 seconds with return value 0
Press any key to continue . . .
```

## 2.5)Adile Naşit

```
C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-3.txt
input-3.txt Adli Dosyanın Acilma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GIRISLERINI YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Naşit, Adile

Naşit, Adile'nin Kevin Bacon Sayısı=Sonsuzdur.

-----
Process exited after 297 seconds with return value 0
Press any key to continue . . .
```

### Örnek 3.

#### 3.1)Bağlantısı Olmayan(1):

```
C:\Users\Onur Koşkan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-2.txt
input-2.txt Adli Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Bale, Christian
Bale, Christian'ın Kevin Bacon Sayısı=Sonsuzdur.
-----
Process exited after 15 seconds with return value 0
Press any key to continue . . .
```

#### 3.2)Bağlantısı Olmayan(2):

```
C:\Users\Onur Koşkan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-1.txt
input-1.txt Adli Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Depp, Johnny
Depp, Johnny'nin Kevin Bacon Sayısı=Sonsuzdur.
-----
Process exited after 15.28 seconds with return value 0
Press any key to continue . . .
```

### 3.3)Bağlantısı Olmayan(3):

C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe

```
Dosya Adını Girin(.txt Formatında) : input-1.txt
input-1.txt Adli Dosyanın Acilma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Gyllenhaal, Jake

Gyllenhaal, Jake nin Kevin Bacon Sayısı=Sonsuzdur.
-----
Process exited after 9.745 seconds with return value 0
Press any key to continue . . .
```

### 3.4)Bağlantısı Olmayan(4):

C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe

```
Dosya Adını Girin(.txt Formatında) : input-2.txt
input-2.txt Adli Dosyanın Acilma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Wood, Elijah

Wood, Elijah nin Kevin Bacon Sayısı=Sonsuzdur.
-----
Process exited after 16.72 seconds with return value 0
Press any key to continue . . .
```



### 3.5)Bağlantısı Olan(1):

```
C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-2.txt
input-2.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):McKellen, Ian
Last Action Hero (1993)
Schwarzenegger, Arnold
Arnold Schwarzenegger: Hollywood Hero (1999)
Coburn, James
Hell Is for Heroes (1962)
Guardino, Harry
Hell with Heroes, The (1968)
McCarthy, Kevin
Hero at Large (1980)
Bacon, Kevin
McKellen, Ian nin Kevin Bacon Sayısı=5.

-----
Process exited after 16.2 seconds with return value 0
Press any key to continue . . .
```

### 3.6)Bağlantısı Olan(2):

```
C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe
Dosya Adını Girin(.txt Formatında) : input-2.txt
input-2.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Asner, Edward
Gargoyles: The Heroes Awaken (1994)
David, Keith
Arnold Schwarzenegger: Hollywood Hero (1999)
Coburn, James
Hell Is for Heroes (1962)
Guardino, Harry
Hell with Heroes, The (1968)
McCarthy, Kevin
Hero at Large (1980)
Bacon, Kevin
Asner, Edward nin Kevin Bacon Sayısı=5.

-----
Process exited after 12.04 seconds with return value 0
Press any key to continue . . .
```

### 3.7)Bağlantısı Olan(3):

C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe

```
Dosya Adını Girin(.txt Formatında) : input-2.txt
input-2.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):Hoffman, Dustin
Hero (1992)
Arnold, Tom
Arnold Schwarzenegger: Hollywood Hero (1999)
Coburn, James
Hell Is for Heroes (1962)
Guardino, Harry
Hell with Heroes, The (1968)
McCarthy, Kevin
Hero at Large (1980)
Bacon, Kevin
Hoffman, Dustin nin Kevin Bacon Sayısı=5.

-----
Process exited after 12.38 seconds with return value 0
Press any key to continue . . .
```

### 3.8)Bağlantısı Olan(4):

C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\PROJE\RAPOR\17011701.exe

```
Dosya Adını Girin(.txt Formatında) : input-1.txt
input-1.txt Adlı Dosyanın Açılma İşlemi Başlatıldı Lütfen Bekleyin...

TUM AKTORLERIN FREKANS LISTESINI OGRENMEK ICIN = 1
KENDİNİZİN BELİRLEYECEĞİ AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2
GİRİSLERİNİ YAPIN:2

Aktor İsmi Girin(Örnek Format=Bacon, Kevin):De Niro, Robert
Sleepers (1996)
Bacon, Kevin
De Niro, Robert nin Kevin Bacon Sayısı=1.

-----
Process exited after 9.685 seconds with return value 0
Press any key to continue . . .
```

## AÇIKLAMALARIYLA BİRLİKTE PROGRAMIN KODLARI:

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#define VERTEX 300000 //VERTEX DEGİSKENİ TEKRAR TEKRAR TANIMLANMAMASI İCİN PROGRAM
BASLANGICINDA 30000 OLARAK MAKRO SEKLİNDE TANIMLANIR
//GEREKLİ STRUCT YAPILARIMIZIN TANIMLANMASI;
struct movies{
    char name[100];
};

struct actors{
    char name[100];
};

struct Qnode{
    int s;
    struct node* next;
};

struct queue{
    struct Qnode* front;
    struct Qnode* rear;
};

struct node{
    int vertex;
    struct node* next;
};

struct Graph {
    int numVertices;
    struct node** adjList;
};

FILE* fp; //İDEN FAZLA FONKSİYON İCERİSİNDE DOSYA İŞLEMLERİ YAPILACAGINDAN DOLAYI GLOBAL
OLARAK TANIMLANIR
//GEREKLİ STRUCT YAPILARIMIZIN BASLANGICTA İNİTİLİAZE EDİLMESİ GEREKTİĞİNDEN BU İŞLEM GLOBAL
OLARAK YAPILIR(ALT TAKI 4 SATIR İCİN);
struct Graph* graph;
struct queue* createQueue(struct queue*);
struct movies filmler[15000];
struct actors oyuncular[200000];
int movieNumber,actorSayisi,movieSayisi = 0,frequency[50] = {0}; //FARKLI FONKSİYONLARDAN DEĞERLERİNİN
AYRI AYRI MUDAHALE EDİLMESİ İCİN
//GEREKLİ DEĞİSKENLERİN GLOBAL OLARAK TANIMLANMASI.

struct Qnode* createNodeQueue(int Index){ //KUYRUGA EKLENİCEK DUGUMU OLUSTURUR;INDEX DEĞERİ ALIR VE
YENİ NODE U DONDURUR.
    struct Qnode* newNode = (struct Qnode*) malloc(sizeof(struct Qnode));
    newNode->s = Index;
    newNode->next = NULL;
    return newNode;
}

struct queue* createQueue(struct queue* kuyruk){ //YENİ KUYRUK OLUSTURMA;KUYRUK TIPİNDE PARAMETRE
ALIR VE KUYRUK TIPİNDE DEĞİSKEN RETURN EDER
    kuyruk = (struct queue*) malloc(sizeof(struct queue));
    kuyruk->front = NULL;
    kuyruk->rear = NULL;

    return kuyruk;
}
```

```
void enqueue(struct queue** kuyruk, int Index){ //KUYRUGA EKLEME ISLEMI;KUYRUK TIPINDE VE INT TIPINDE  
INDEX ALIR.RETURN DEGERI YOKTUR(VOID FONKSIYON)
```

```
    struct Qnode* newNode = createNodeQueue(Index);
```

```
    if((*kuyruk)->rear == NULL){  
        (*kuyruk)->rear = newNode;  
        (*kuyruk)->front = newNode;  
    }  
    else{  
        (*kuyruk)->rear->next = newNode;  
        (*kuyruk)->rear = newNode;  
    }  
}
```

```
}
```

```
int dequeue(struct queue** kuyruk){ //KUYRUKTAN CEKME ISLEMI;KUYRUK TIPINDE PARAMETRE ALIR,INT  
TIPINDE INDIS DONDURUR
```

```
    if((*kuyruk)->front == NULL){  
        return -3;  
    }  
    struct Qnode* temp;  
    int x = (*kuyruk)->front->s;  
    temp = (*kuyruk)->front;  
    (*kuyruk)->front = (*kuyruk)->front->next;  
    if((*kuyruk)->front == NULL){  
        (*kuyruk)->rear = NULL;  
    }  
    free(temp);  
    return x;  
}
```

```
int isEmpty(struct queue* queue){ //GELEN KUYRUK TIPLI PARAMETRELİ KUYRUK MEVCUTMU?INTEGER  
TIPINDE DEGER DONDURUR.
```

```
    if(queue->front == NULL)//BULUNMUYOR ISE 1 DONDURUR  
        return 1;
```

```
    else //KUYRUK MEVCUT ISE 0 DONDURUR  
        return 0;
```

```
}
```

```
struct node* createNode(int v){ //NODE OLUSTURULUR;INT TIPINDE PARAMETRE ALIR VE NODE TIPINDE DEGER  
DONDURUR.
```

```
    struct node* newNode;  
    newNode = (struct node*) malloc(sizeof(struct node));  
    newNode->vertex = v;  
    newNode->next = NULL;  
    return newNode;  
}
```

```
struct Graph* createGraph(int nVertex){ //GRAPH OLUSTURULUR;INT TIPINDE PARAMETRE ALIR VE GRAPH TIPINDE  
DEGER DONDURUR.
```

```
    int i;  
    struct Graph* graph = (struct Graph*) malloc(sizeof(struct Graph));  
    graph->numVertices = nVertex;  
    graph->adjList = (struct node**) malloc(nVertex*sizeof(struct node*));  
    for (i = 0; i < nVertex ;i++) {  
        graph->adjList[i] = NULL;  
    }  
    return graph;  
}
```

```
void addEdge(struct Graph* graph, int source, int destination){ //EDGE EKLENİLİR;GRAPH 1,INTEGER OLMAK UZERE  
2(KAYNAK VE HEDEF) PARAMETRE ALIR.DONUS TIPI OLMAZ(VOID)
```

```
    struct node* newNode = createNode(destination);
```

```

newNode->next = graph->adjList[source];
graph->adjList[source] = newNode;
newNode = createNode(source);
newNode->next = graph->adjList[destination];
graph->adjList[destination] = newNode;
}

void BFSforAll(struct queue** kuyruk, int startingIndex){ //TUM AKTORLERIN KEVIN BACON SAYISINI BULAN
FONKSIYON;KUYRUK TIPINDE VE
//INTEGER(BASLANGIC INDISI) TIPINDE PARAMETRE ALIR.DONUS DEGERI YOKTUR.
int i,v;
struct node* temp;
int visited[VERTEX];
for(i = 0; i < VERTEX; i++) //AKTORLERE BASLANGIC DEGERI ATANIR
    visited[i] = -1;

enqueue(kuyruk,startingIndex); //ILK GELEN DUGUM EKLENIR
visited[startingIndex] = 0;

while(!isQueueEmpty((*kuyruk))){
    v = dequeue(kuyruk); //KUYRUKTAN CEKILME ISLEMI
    temp = graph->adjList[v]; //KOMSU DUGUMLERIN ISLEME ALINMASI
    while(temp != NULL){
        if(visited[temp->vertex] == -1){//DAHA ONCE GEZILMEYEN DUGUMLER(AKTORLER) BULUNUR
            enqueue(kuyruk,temp->vertex); //KUYRUGA EKLENIR
            visited[temp->vertex] = visited[v] + 1;
        }
        temp = temp->next; //ITERASYON ISLEMI
    }
}

for(i = movieNumber; i < actorSayisi; i++){ //FILM SAYISINDAN BASLAR->AKTOR SAYISI KADAR ILERLER
    if(visited[i] != -1) //DAHA ONCE GEZILMEDİYSE
        frequency[visited[i]/2]++; //SADECE AKTORLERLE ISLEM YAPMAK ICIN FILMLERİ ES GECEMEMİZ
        GEREK.BUNDAN DOLAYI MEVCUT INDISIN YARISINDAN ISLEM YAPILIR.

    else
        frequency[49]++; //KEVIN BACON SAYISININ SONSUZ OLANLARIN HESAPLANMASI
}

for(i=0;i < 49; i++){//KEVIN BACON SAYILARININ EKRANA YAZDIRILMASI;
    if(frequency[i]>0)
        printf("%d Kisinin Kevin Bacon Sayisi = %d.\n", frequency[i], i);
}
// SONSUZ KEVIN BACON SAYISI OLANLARIN EKRANA YAZDIRILMASI.
printf("%d Kisinin Kevin Bacon Sayisi = Sonsuz", frequency[49]);
}

void graphDoldur(struct Graph* graph, char* filename){ //GRAPHI DOLDURAN FONKSIYON;GRAPH TIPINDE VE CHAR
STRING TIPINDE(DOSYA ADI ICIN) DEGİSKEN ALIR DONUS DEGERI YOKTUR.
char* token,*film,*aktor,string[5000];
int keep; //MEVCUT INDISDEKI DEGERIN KONTROLU ICIN KULLANILAN DEGİSKEN

fp = fopen(filename,"r");
printf("%s Adli Dosyanin Acilma Islemi Baslatildi Lutfen Bekleyin...\n", filename);
if(fp == NULL) //DOSYA ACILAMADIYSA
    printf("Dosya Acilamadi!");

else{ //DOSYA ACILDIYSA
    while(fgets(string,5000,fp)){ //ILK 5000 KELİMEYİ OKUR STRING DİZİSİNE ATAR(5000 SAYISI STRING
DİZİSİNİN BOYUTUNDAN DOLAYIDIR)
        //ALT TAKI 3 SATIRDA KULLANILAN STRTOK FONKSIYONLARINDA ILK OPERANDA OKUMA
YAPILIR,İKİNCİ OPERANDDAKİ DEĞER REFERANSI İLE PARCALAYARAK OKUYOR
        token = strtok(string,"\n");
        film = strtok(token,"/");

```

```

    aktor = strtok(NULL, "/");
    strcpy(filmler[movieSayisi].name, film); //IKINCI OPERAND,ILK OPERANDA KOPYALANIR
    while(aktor != NULL){
        keep = checkIndex(aktor); //AKTORUN MEVCUTLUK KONTROLU(MEVCUT ISE INDIS TUTAR,MEVCUT
DEGIL ISE -1 TUTAR)
        if(keep == -1){ //AKTOR DAHA ONCEDEN OKUNMADIYSA YENI OLUSTURUYOR VE EDGE EKLENİYOR.
            strcpy(oyuncular[actorSayisi-movieNumber].name,aktor);
            addEdge(graph,movieSayisi,actorSayisi);
            actorSayisi++;
        }
        else{ //OKUNDUYSA EDGE EKLIYOR
            addEdge(graph,movieSayisi,keep);
        }
        aktor = strtok(NULL, "/");
    }
    movieSayisi++;
}
}

fclose(fp);
}

int checkIndex(char* string){ //GONDERILEN CHAR STRINGIN VARLIGININ KONTROLU
    int i=0;
    while(i<actorSayisi-movieNumber){
        if(!strcmp(oyuncular[i].name,string)) //AKTOR MEVCUT ISE INDISINI DONDURUYOR
            return movieNumber + i;
        i++;
    }
    return -1; //MEVCUT DEGIL ISE
}

int BFS(struct queue** kuyruk, int startingIndex, int index, int type){ //SPESIFIK OLARAK PARAMETRESI GONDERILEN
AKTORE BREADTH FIRST SEARCH YAPILMASI;
//KUYRUK TIPINDE 1 VE INTEGER TIPINDE 3 PAREMETRE ALIR.INTEGER TIPINDE DEGER DONDURUR.
    if(startingIndex == index) //GIRILEN AKTOR,MEVCUT INDISDE ISE KENDISINE ESITTIR VE KEVIN BACON
SAYISI ODIR
        return 0;

    else{ //EGER DEGILSE
        int i,v;
        //FONKSIYON ICERISINDEKI DEGERLERINI KAYBETMEMELERI ICIN STATIC TANIMLANAN
DEGISKENLERIMIZ;
        static counter = 0;
        static int visited[VERTEX];
        static char* name;

        struct node* temp;
        for(i = 0; i < VERTEX; i++) //AKTORLERE BASLANGIC DEGERI ATANIR(DAHA ONCEDEN GEZILIP
GEZILMEDIGINI KONTROL ETMEK ICIN).
            visited[i] = -1;

        enqueue(kuyruk,startingIndex); //KUYRUGA EKLENIR
        visited[startingIndex] = 0;

        while(!isEmpty(*kuyruk)){ //KUYRUGUN BOS OLUP OLMAMASININ KONTROLU
            v = dequeue(kuyruk);
            temp = graph->adjList[v];

            while(temp != NULL){ //BOS OLMADIKCA ITERASYONA DEVAM EDILIR;
                if(visited[temp->vertex] == -1){
                    enqueue(kuyruk,temp->vertex);
                    visited[temp->vertex] = visited[v] + 1;

                    if(temp->vertex == index){
                        int tempp = visited[temp->vertex]/2;

```

```

        if(type == 2)
            return tempp;

//V DEGISKENINDEKI MEVCUT DEGER FILM SAYISINDAN BUYUK VEYA ESIT ISE;
//OYUNCUYU YAZDIR,DEGIL ISE FILMI YAZDIR;
(v>=movieNumber?printf("%s\n",oyuncular[v-movieNumber].name):printf("%s\n",filmler[v].name));

//KUYRUGU YENIDEN OLUSTURMA;
while(!isEmpty(*kuyruk))
    dequeue(kuyruk);
*kuyruk = createQueue(*kuyruk);

counter++;
if(v != startingIndex) //MEVCUT INDISDE DEGIL ISE
    return BFS(kuyruk,startingIndex,v,1); //BREADTH FIRST SEARCH ISLEMINE RECURSIVE OLARAK
DEVAM EDILIR

```

```

        else
            return counter/2; //MEVCUT INDISE VARMIS ISE AKTOR VE FILMLER 2 KERE SAYILDIGI ICIN
YARISI RETURN EDILIR.

```

```

    }
}
temp = temp->next; //ITERASYON ISLEMI
}
}
return -1;
}
}

```

```

int main(){
    int select, indexToStart, indexToFind, kevinBaconSayisi;
    char name[50],filename[15],str[5000];
    struct queue* kuyruk;
    kuyruk = createQueue(kuyruk);
    graph = createGraph(200000);

    printf("Dosya Adini Girin(.txt Formatinda) : ");
    gets(filename);

    fp = fopen(filename,"r");
    if(fp == NULL)//DOSYA ACILAMADI,PROGRAM KAPATILIR.
        printf("Dosya acilirken hata olustu!");

    else{//DOSYA ACILDI
        //FILM SAYISINI OGRENME;
        while(fgets(str,4099,fp)) //MEVCUT BOYUTUN -1 KADARINA KADAR GİTMELİ /0'İ ALMAMASI ICIN
            movieNumber++;

        fclose(fp);

        actorSayisi = movieNumber;
        graphDoldur(graph,filename);
        indexToStart = checkIndex("Bacon, Kevin");

        printf("\nTUM AKTORLERIN FREKANS LİSTESİNİ OGRENMEK ICIN = 1\nKENDİNİZİN BELİRLEYECEĞİ
AKTORUN KEVIN BACON SAYISINI OGRENMEK ICIN = 2\nGİRİSLERİNİ YAPIN:");
        scanf("%d",&select);

        if(select == 1)
            BFSforAll(&kuyruk, indexToStart);    //(1)

        else if(select == 2){
            printf("\nAktor İsmi Girin(Ornek Format=Bacon, Kevin):");

```

```
scanf(" %[^\\n]s",name);

indexToFind = checkIndex(name);
kevinBaconSayisi = BFS(&kuyruk, indexToStart, indexToFind,1);    //(2)
if(kevinBaconSayisi > 0)
    printf("%s nin Kevin Bacon Sayisi=%d.\\n",name,kevinBaconSayisi);

else if(kevinBaconSayisi == 0)
    printf("Girilen Aktor Kevin Bacondur.Dolayisiyla Kevin Baconun,Kevin Bacon Sayisi 0dir.\\n");

else
    printf("\\n%s nin Kevin Bacon Sayisi=Sonsuzdur.",name);

}
else
    printf("Lutfen Sadece 1 Veya 2 Girislerini Kullanim.");
}
return 0;
}
```