

**Amac****Graph****Soru 1**

Bir dugumun komsuluk bilgilerini tutabileceginiz struct AdjListNode yapisini ve komsuluk listelerini tutacaginiz yapiyi Struct AdjList olusturunuz.

**Soru 2**

Komsuluk bilgisinden yola cikarak olusturacaginiz Graph'in yapisini struct Graph olusturunuz.

**Soru 3**

Biur dugume yeni bir komsuluk listesi olusturacak olan struct AdjListNode\* newAdjListNode(int dest) fonksiyonu yaziniz.

**Soru 4**

Yaratilmek istenilen graph'in kose sayisini parametre ile alan ve graphi olusturan struct Graph\* createGraph(int V) fonksiyonunu yaziniz.

**Soru 5**

Verilen graph a kenar ekleyen void addEdge(struct Graph\* graph, int src, int dest) fonksiyonunu yaziniz.

**Soru 6**

Ekte verilen sensor konumlarini tutabilmek icin uygun yapiyi olusturunuz.

**Soru 7**

Ekte verilen sensor konumlarini dosyadan cekip yukarida belirttiginiz yapi seklinde bir dizide (array of structure) saklayan fonksiyonu yaziniz.

**Soru 8**

SensorID'si 0 olan sensoru merkez dugum olarak kabul edin ve haberlesme mesafesinin 30m oldugu bir ortama gore komsuluk matrisini olusturun.

## Soru 9

Bir önceki soruda oluşturdugunuz komsuluk matrisini temel alarak her dugumun kacinci dereceden bir dugum oldugunu (merkez dugume olan uzakligi) hesaplayan fonksiyonu yaziniz.

## Soru 10

Haberlesme mesafesi 30 metre icin her sensor icin “SensorID: XX (x,y), Merkez Dugume olan uzakligi: X, Komsu sayisi: X, Komsulari={X,X,X,X}” seklinde bir cikti veriniz

## Soru 11

Haberlesme mesafesini once 20 sonra 10 metre yaparak Soru 11’deki islemleri tekrarlayiniz. Mesafenin azligindan dolayi haberlesilemeyen dugumleri belirtiniz. Ornek: 10 metre icin “SensorID: XX (x,y), Merkez Dugume olan uzakligi: X, Komsu sayisi: 0, Komsulari={} Not: Bu sensorun hic komsusu yoktur. Izole dugum” seklinde bir cikti veriniz