

BLM226 NESNEYE DAYALI PROGRAMLAMA

PROJE

Teslim zamanı: 21/04/2020 Salı 20:00

BİR GRUP İNSAN İÇİNDE ÜNLÜ OLAN KİŞİYİ BULMAK:

N kişinin katıldığı bir toplantı ve toplantıdaki herkesin tanıdığı yalnız bir kişi var. Bu kişi toplantıda bulunabilir de, bulunmayabilir de. Tanınan kişi toplantıda bulunuyorsa, toplantıdaki diğer kişilerin hiçbirini tanımıyor.

Yalnızca, “ A kişisi B kişisini tanıyor mu?” türünde sorular sorabilir ve cevabını alabiliriz. Çözmemiz gereken problem, herkesin tanıdığı ünlü kişinin kim olduğunu en az soruyla bulmak.

Bu problemi çözmenin bir yolu, yığın (stack) veri yapısını kullanmak. Algoritma şöyle çalışıyor:

1. Bütün kişileri bir yığına it (push).
2. Yığından 2 kişi çek (pop). “ A , B ’yi tanıyor mu?” sorusunun cevabına bağlı olarak, iki kişinin birinden kurtul (çöpe at).
3. Kalan kişiyi yığına it (push).
4. Yığında bir kişi kalana kadar 2. ve 3. adımları yinele.
5. Yığında kalan kişinin başka kişileri tanımadığını kontrol et.

Anlatılan çözümde, en fazla N kişiyi çöpe atabiliriz. Ünlü kişi toplantıda bulunuyorsa, *DOĞRU* (*TRUE*) veya *YANLIŞ* (*FALSE*) dönen *taniyor?(A, B)* fonksiyonunu $3(N-1)$ kere çağırırız.

Girdi ve çıktı

Programınıza girdi olarak bir metin dosyasının adı verilecek ve dosyadaki veriler okunacak. Bu dosyanın, ilk satırında toplantıdaki kişi sayısı bir tamsayı olarak, sonraki satırlarında ise hangi kişinin hangi kişiyi tanıdığına ilişkin bilgiler tamsayı çiftleri halinde bulunacak. Programınızın çıktısı, herkesin tanıdığı kişinin kimliğini veya, öyle birisi yoksa, bu durumu bildirmeli.

Aşağıdaki örnek *girdi.txt* dosyasını inceleyelim.

girdi.txt

```
4
0 2
1 2
3 2
0 1
1 0
3 1
```

Dosyadaki veri bize şunları söylüyor:

- Toplantıda 4 kişi var.
- 0. kişi 2. kişiyi tanıyor.
- 1. kişi 2. kişiyi tanıyor.
- 3. kişi 2. kişiyi tanıyor.
- 0. kişi 1. kişiyi tanıyor.
- 1. kişi 0. kişiyi tanıyor.
- 3. kişi 1. kişiyi tanıyor.

girdi.txt dosyasındaki verilere göre ünlü kişinin 2. kişi olduğunu görüyoruz.

Bir de aşağıdaki örnek *girdi2.txt* dosyasına bakalım:

girdi2.txt

```
3
0 2
1 0
2 1
```

Bu dosyadaki veri de şöyle:

- Toplantıda 3 kişi var.
- 0. kişi 2. kişiyi tanıyor.
- 1. kişi 0. kişiyi tanıyor.
- 2. kişi 1. kişiyi tanıyor.

girdi2.txt dosyasındaki verilere göre toplantıda ünlü kişinin olmadığını görüyoruz.

Örnek koşmalar:

Yukarıda verilen *girdi.txt* ve *girdi2.txt* dosyaları üzerinde program aşağıdaki gibi koşmalı:

```
Dosya adi girin: girdi.txt
Unlu kisi 2 numarali kisi!
```

```
Dosya adi girin: girdi2.txt
Unlu kisi yok.
```

Konsoldan (klavyeden) girdi okumak için kod parçası:

```
using System;
...
string str;
str = Console.ReadLine();
```

Metin dosyasından (*girdi.txt*) girdiyi okumak için kullanılabilecek örnek kod parçası:

```
using System.IO;
...
FileStream fin = new FileStream("girdi.txt", FileMode.Open);
StreamReader fstr_in = new StreamReader(fin);

int i, a, b;
string temp;
string [] tokens;

temp = fstr_in.ReadLine(); //Ilk satirdaki bir sayiyi okuma
i = int.Parse(temp);

//Sonraki satirlardaki ikişer sayiyi okuma

while ((temp = fstr_in.ReadLine()) != null) {
    tokens = temp.Split();
    a = int.Parse(tokens[0]);
    b = int.Parse(tokens[1]);

    //Okunan iki sayıyla yapılacak işler
    ...
}

//Kodun kalanında yapmak istedikleriniz
...
```

UYARILAR, PROJENİN TESLİM EDİLMESİ, EK BİLGİLER:

- Girdi dosyalarının doğru formatta olduğunu ve hata bulunmadığını varsayabilirsiniz.
- Programı geliştirdiğiniz çalışma uzayı (kaynak kodlar, vs.) ve ne yaptığınızı anlatan proje dokümanı teslim edilecek. Kod dosyaları birden fazla olabilir, ama doküman tek bir dosya (*.docx veya *.pdf) olmalı. Teslim etmeniz gereken dosyaları “B226p_” diye başlayıp kendi adınızla biten adı olan bir klasöre koyun. Bu klasörü sıkıştırın (.zip veya .rar) ve “Nesneye Dayalı Programlama” dersine ait Google classroom’a yükleyin.

Sıkıştırılmış dosya adı örnekleri:

- B226p_ayse_sari.zip
- B226p_mehmet_ali_tok.rar
- Proje dokümanının nasıl yazılacağına ilişkin ipuçlarını, “Nesneye Dayalı Programlama” dersine ait Google classroom’da yayınlanan yazılım tasarım belgesi şablonunda (ytb.pdf) bulabilirsiniz. Yazılım tasarım belgesi şablonu, size yol gösterme amaçlıdır. Yaptığınız çalışmayla ilgili görmediğiniz bölümleri iptal edebilir, ilgili olduğunu düşündüğünüz bölümleri ekleyebilirsiniz. Anlaşılır olmak koşuluyla, kendi tarzınız konusunda özgürsünüz.
- Proje dokümanına, örnek girdi ve çıktıları (en az 2 tane) eklemeyi unutmayın.
- Kodunuzun doğruluğu kadar, okunabilirlik ve dokümantasyon da önemlidir.