



VERİ YAPILARI 2021-2022 Bahar Dönemi Ödev- 2

Ödev Duyurulma Tarihi : 17 Mayıs 2022 Salı

Ödev Teslim Tarihi : 27 Mayıs 2022, Cuma 23:59

1. TANIMLAMA

Bu ödevde *İyilik Yolum* isimli sosyal yardımlaşma platformu mobil uygulamasının arka planında yapılması istenen işlemleri belli kurallar dâhilinde gerçekleştiren C++ kodunu yazmanız gerekmektedir. Kodu yazarken heap (öncelikli kuyruk) ve graf veri yapısı kullanılacaktır. Ayrıca ihtiyaç dahilinde diğer veri yapılarını da kullanabilirsiniz. (C++ dilinde hazır **STL kütüphaneleri** kullanılmayacaktır.)



Şekil 1. Harita

İyilik Yolum isimli mobil uygulamasının senaryosu aşağıdaki şekildedir:

Bu uygulama iyilik yapmak isteyen kullanıcının kendi aracı ile bir yerden başka bir yere giderken yol üstünde araca ihtiyacı olan kişiler (yaşlı, hasta, engelli, öğrenci, hamile, bebek sahibi kadın) var ise gönüllü olarak bu kişileri yoldan alarak gideceği yere götürmesini sağlamaktadır. Bu sayede yardım etmek isteyen kişilerin ihtiyaç sahiplerine hızlı bir şekilde ulaşması amaçlanmaktadır.

İyilik Yolum isimli uygulamada iki türlü kullanıcı girişi yapılmaktadır.

1. **İyilik Yap Yoldan Al:** Kullanıcı harita üzerinde bir noktadan diğer noktaya gideceği yolu belirler. Bu yol üzerinde araca ihtiyacı olan kişileri görebilir. Yardımsever kullanıcı, ihtiyaç sahibi kişileri belirlenen noktalardan alarak önceliklerine göre gidecekleri yerlere götürür.

2. **Yoldayım Beni Al:** İhtiyaç sahibi kullanıcı harita üzerinde mevcut konumunu işaretleyerek yardımsever bir kullanıcının kendisini alması için bekler.

2. PROBLEM

İyilik Yolum isimli uygulama iki bölümden oluşmaktadır. Yoldayım Beni Al bölümünde ihtiyaç sahibi kullanıcı *mevcut konumu (bekleme noktası)*, *öncelik türünü* ve *gitmek istediği konum (hedef)* bilgisini sisteme girer ve beklemeye başlar.

İyilik Yap Yoldan Al bölümünde ise yardımsever kullanıcı ilk olarak *mevcut konum (başlangıç noktası)* bilgisi ile *gideceği konum (varış noktası)* bilgisini sisteme girer. Böylelikle kullanıcının gideceği en uygun rota belirlenir. Yardımsever kullanıcı belirlenen bu yol üzerinde ihtiyaç sahiplerinin bulundukları konum ve gitmek istedikleri konum bilgilerini haritada görebilir.

Bu bilgiler ile yardımsever kullanıcının varış noktasına ulaşmaya kadar kaç kişiye yardım edebileceği belirlenecektir.

Problem adımları:

1. Yardımsever kullanıcıların başlangıç noktaları ile varış noktaları Tablo 1’de verilmiştir. Bu bilgiler haritadan okunarak her kullanıcı için başlangıç noktasından varış noktasına ulaştıracak en uygun rota belirlenecektir.
2. Yardımsever kullanıcıların rotaları üzerinde ihtiyaç sahibi kullanıcıların olup olmadığı belirlenecektir.
3. Her yardımsever kullanıcının aracı 5 kişilik olup, kendisi hariç 4 yolcu daha alabilecektir.
4. Yardımsever kullanıcı rotası üzerinde harekete başlar, geçtiği noktada ihtiyaç sahibi kullanıcı bekliyorsa onu aracına alır, ihtiyaç sahibi kullanıcının gideceği noktaya göre yeniden en uygun rota belirlenir. İhtiyaç sahibi hedef noktasına bırakılınca yardımsever kullanıcının varış noktasına göre yeniden rota belirlenir.
5. Rota üzerindeki bir noktada birden fazla ihtiyaç sahibi kullanıcı bekliyorsa, yardımsever kullanıcı aracının kapasitesine ve bekleyen kişilerin öncelik puanlarına göre ihtiyaç sahibi kullanıcıları aracına alır. İhtiyaç sahibi kullanıcıların hedef noktalarına hangi sırayla gidecekleri öncelik puanlarına göre belirlenir. İlk olarak öncelik puanı en yüksek olan ihtiyaç sahibi için rota belirlenir ve gideceği hedef noktasına götürülür. Daha sonra sırayla tüm yolcular hedef noktalarına götürülür.
6. Her belirlenen yeni rota üzerinde bekleyen ihtiyaç sahipleri bulunuyorsa 4. ve 5. maddelerdeki işlemler tekrar edilir.
7. İhtiyaç sahiplerinin öncelik türleri ve öncelik puanları Tablo 2’de verilmiştir. İhtiyaç sahiplerinin bekleme noktaları ile hedef noktaları ise Tablo 3’te verilmiştir.
8. Yardımsever kullanıcı 4 yolcu almış bir şekilde rota üzerinde ilerlerken bekleyen ihtiyaç sahibi ile karşılaşır;

- a. Bekleyen ihtiyaç sahibi kullanıcının öncelik puanı, araç içindeki yolcuların birinden daha yüksek ise, en düşük öncelik puanına sahip yolcu araçtan indirilir, önceliği yüksek olan ihtiyaç sahibi araca alınır. Araçtaki yolcuların öncelik puanlarına göre yeniden rota belirlenir. Araçtan indirilen yolcunun öncelik puanı 10 puan artırılır, yolcu yeniden beklemeye başlar.
- b. Araçta eşit öncelik puanına sahip yolcular var ise Tablo 2’ de verilen öncelik türüne göre sıralama yapılır, öncelik türü de aynı ise en son araca alınan yolcu araçtan indirilir, öncelik puanı 10 puan artırılır.
- c. Bekleyen ihtiyaç sahibi kullanıcının öncelik puanı, araç içindeki yolcuların hepsinden daha düşük ise bekleyen kişi araca alınmaz beklemeye devam eder ancak öncelik puanı 5 puan artırılır.
9. Araçta eşit öncelik puanına sahip yolcular var ise Tablo 2’ de verilen öncelik türüne göre sıralama yapılır, öncelik türü de aynı ise daha önce araca alınan yolcu ilk olarak gideceği hedef noktasına bırakılır.
10. Hedef noktasına bırakılmış öncelik sahibi yolcu tekrar araca alınmaz, sistemden çıkarılır.

Tablo 1. Yardımsever Kullanıcıların Başlangıç ve Varış Noktaları

	Başlangıç Noktası	Varış Noktası
Yardımsever Kullanıcı 1 (YSK 1)	(10,0)	Restoran (23,23)
Yardımsever Kullanıcı 2 (YSK 2)	(11,24)	Kafe (1,1)

Tablo 2. İhtiyaç Sahibi Öncelik Bilgileri

Sıra	Öncelik Türü	Öncelik Puanı
1	Doğum Yapmak Üzere Olan Hamile	60
2	Çok Yaşlı Erkek/Kadın	25
3	Doktor Kontrolüne Gidecek Hasta	20
4	Engelli Erkek/Kadın	15
5	Kucağında Bebeği Olan Kadın	10
6	Öğrenci	5

Tablo 3. İhtiyaç Sahiplerinin Bekleme ve Hedef Noktaları Bilgileri

No	Yolcu Türü	Bekleme Noktası	Hedef Noktası
1	Engelli 1	Hastane 1	Eczane 1
2	Bebekli Kadın 1	Hastane 1	Market 2
3	Yaşlı 1	Hastane 1	Hastane 2
4	Öğrenci 1	Okul 1	Market 1
5	Öğrenci 2	Okul 1	Eczane 2
6	Öğrenci 3	Okul 1	Hastane 2
7	Hamile 1	Eczane 1	Hastane 2
8	Hasta 1	Hastane 2	Eczane 2
9	Bebekli Kadın 2	Hastane 2	Market 2
10	Yaşlı 2	Hastane 2	Okul 2
11	Hamile 2	Eczane 2	Hastane 1
12	Engelli 2	Eczane 2	Okul 1
13	Bebekli Kadın 3	Market 2	Hastane 1
14	Hamile 3	Eczane 2	Hastane 2
15	Öğrenci 4	Market 2	Okul 2

3. OLUŞTURULACAK SINIFLAR

(Kodu yazarken ihtiyaç durumunda daha fazla sınıf ve fonksiyon oluşturulabilir.)

main.cpp

haritaOku() : **harita.txt** dosyasından harita bilgileri okunacaktır (Bkz Tablo-1).

yardimseverRotaBul():Yardımsaver kullanıcı haritada başlangıç noktasından itibaren gezinmeye başlar, varış noktasına ulaşmak için en uygun yol seçimini yapar. Yardımsaver kullanıcı Rota üzerinde ilerlerken ihtiyaç sahibi kullanıcı ile karşılaşınca onu aracına alır ve yeni rota belirlenir. Önce ihtiyaç sahibinin gideceği hedef noktaya gidilir, ihtiyaç sahibi kullanıcı araçtan indirildikten sonra tekrar varış noktası için rota belirlenir. Varış noktasına ulaşmaya kadar rota üzerinde karşılaşılan tüm ihtiyaç sahibi kullanıcılar için araca alma, yeni rota belirleyip hedefe ulaşma, yeniden varış noktasına doğru rota belirleme işlemleri tekrar edilir. Yardımsaver kullanıcıların gezindiği her düğüm bilgisi **sonuc.txt** dosyasına yazılmalıdır (yanlış gidilip tekrar geri dönülen yollar dâhil). Varış noktasına ulaşmak için herhangi bir yön değişiminde, önce yön bilgisi (sağ, sol, yukarı ve aşağı) yazılıp daha sonra düğümlerin yazımına yeni satırdan itibaren başlanır.

ihtiyacSahibiniYoldanAl(): Yardımsaver kullanıcı bir noktadan geçerken ihtiyaç sahibinin öncelik puanına bakar, araçta birden fazla ihtiyaç sahibi kullanıcı var ise veya bekleme noktasında birden fazla ihtiyaç sahibi ile karşılaşır ise öncelik puanına göre ihtiyaç sahiplerini araca alır, sıralama yapar ve hedef noktalarına bırakır.

ihtiyacSahibiniBeklet(): Yardımsaver kullanıcı 4 yolcu almış bir şekilde rota üzerinde ilerlerken bekleyen ihtiyaç sahibi ile karşılaşır ise, bekleyen ihtiyaç sahibi kullanıcının öncelik puanı, araç içindeki yolcuların birinden daha yüksek ise, en düşük öncelik puanına sahip yolcu araçtan indirilir, önceliği yüksek olan ihtiyaç sahibi araca alınır. Araçtaki yolcuların öncelik puanlarına göre yeniden rota belirlenir. Araçtan indirilen (henüz hedef noktasına varmamış) yolcunun öncelik puanı 10 puan arttırılır, yolcu yeniden beklemeye başlar. Bekleyen ihtiyaç sahibi kullanıcının öncelik puanı, araç içindeki yolcuların hepsinden daha düşük ise bekleyen kişi araca alınmaz beklemeye devam eder ancak öncelik puanı 5 puan arttırılır.

ihtiyacSahibiniHedefeBirik () : Yardımsaver kullanıcıların hangi ihtiyaç sahibini hangi hedef noktasına götürdüğü bilgisi **hedef.txt** dosyasına yazdırılır (Bkz Şekil 5).

4. GİRDİ VERİ DOSYALARI

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	4	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
2	0	1	1	1	1	1	0	1	0	1	0	6	0	0	0	0	0	0	0	1	1	1	1	1	0
3	0	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	0	0	0	0	0	1	0
4	0	1	1	1	1	1	8	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
5	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	10	0	1	1	1	1	1	0
6	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	0
7	0	1	1	1	1	1	0	1	0	1	1	0	0	1	1	1	0	1	0	1	1	1	1	1	0
8	0	1	1	1	1	1	0	1	0	1	1	0	0	1	1	1	0	1	0	1	1	1	1	1	0
9	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	0
10	2	1	1	1	1	1	1	1	0	11	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	3
12	0	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	0
13	0	1	1	1	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0
14	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	0	0	0	0	0
15	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0
16	0	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0
17	0	1	1	1	1	1	0	7	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	0
18	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0
19	0	1	1	1	1	1	1	1	0	1	0	0	0	0	0	1	1	12	1	1	1	1	1	1	0
20	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
21	0	1	1	1	0	1	1	13	0	1	1	1	1	1	1	1	9	0	1	1	1	1	1	1	0
22	0	1	1	1	0	1	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0
23	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	5	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Şekil.2 Haritanın Dijital Gösterimi

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	YSK1:2
0	4	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	YSK2:3
0	1	1	1	1	1	0	1	0	1	0	6	0	0	0	0	0	0	1	1	1	1	1	1	1	0	Kafe:4
0	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	0	0	0	0	0	0	1	0	Restoran:5
0	1	1	1	1	1	8	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	Hastane1:6
0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	10	0	1	1	1	1	1	1	1	1	0	Hastane2:7
0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	Eczane1:8
0	1	1	1	1	1	0	1	0	1	1	0	0	1	1	1	0	1	1	1	1	1	1	1	1	0	Eczane2:9
0	1	1	1	1	1	0	1	0	1	1	0	0	1	1	1	0	1	0	1	1	1	1	1	1	0	Okul1:10
0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0	Okul2:11
2	1	1	1	1	1	1	1	0	11	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	Market1:12
0	0	0	0	0	0	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	3	Market2:13
0	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0	
0	1	1	1	1	1	0	1	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	
0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	0	0	0	0	0	0	
0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0	
0	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0	
0	1	1	1	1	1	0	7	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0	
0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	
0	1	1	1	1	1	1	1	0	1	0	0	0	0	0	0	1	12	1	1	1	1	1	1	1	0	
0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
0	1	1	1	0	1	1	13	0	1	1	1	1	1	1	1	9	0	1	1	1	1	1	1	1	0	
0	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0	
0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	5	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Şekil. 3 harita.txt Dosyasının İçeriği

5. YAPILACAKLAR

1. **harita.txt** dosyasından oyun alanı bilgileri okunacaktır.
2. Yardımsever kullanıcıların varış noktasına ulaşması için gidilen yollar *yardımseverRotaBul()* fonksiyonunda belirtildiği şekilde **sonuc.txt** dosyasına yazdırılacaktır. (Bkz Şekil 4).
3. Haritada gezinirken kapalı olan yollar belirtilmelidir.
4. Yardımsever kullanıcıların hangi ihtiyaç sahibini hangi hedef noktasına götürdüğü bilgisi “İhtiyaç Sahibi? – Öncelik Puanı: – Hedef Noktası (,) – Yardımsever Kullanıcı?” formatında **hedef.txt** dosyasına yazdırılır. (Bkz Şekil 5)

Yardımsever Kullanıcı 1:

(10,0) → (10,1) → (10,2) → (10,3) → (10,4) → (10,5) → (10,6) → (10,7) → aşağı

Araçtaki Yolcu Sayısı: 0

(11,7) → (12,7) → (13,7) → (14,7) → (15,7) → (16,7) → (17,7) → ..

Araçtaki Yolcu Sayısı: 3

--

Araçtan (Öğrenci 1/ Engelli 1 /..vs) Yolcu İndirildi.

Araçtaki Yolcu Sayısı: ...

--

Yardımsever Kullanıcı 1 Araca Aldığı Toplam Yolcu Sayısı: ..

Yardımsever Kullanıcı 1 Varış Noktasına En Kısa Yol: (10,0) → (10,1) → (10,2) → ... → (23,23)

Yardımsever Kullanıcı 2:

(11,24) → (11,23) → (11,22) → (11,21) → (11,20) → (11,19) → (11,18) → (11,17) → yukarı

Araçtaki Yolcu Sayısı: 0

(11,17) → (10,17) → (9,17) → (8,17) → (7,17) → (6,17) → (5,17) → ..

Araçtaki Yolcu Sayısı: 3

--

Araçtan (Öğrenci 1/ Engelli 1 /..vs) Yolcu İndirildi.

Araçtaki Yolcu Sayısı: ...

--

Yardımsever Kullanıcı 2 Araca Aldığı Toplam Yolcu Sayısı: ..

Yardımsever Kullanıcı 2 Varış Noktasına En Kısa Yol: (11,24) → (11,23) → (11,22) → ... → (1,1)

Şekil 4. sonuc.txt dosyasının içeriği

Yaşlı 2 – Öncelik Puanı: 25 – (10,9) – Yardımsever Kullanıcı 1

Hasta 1 – Öncelik Puanı: 20 – (21,16) – Yardımsever Kullanıcı 1

--

Öğrenci 1 – Öncelik Puanı: 5 – (19,17) – Yardımsever Kullanıcı 2

Öğrenci 2 – Öncelik Puanı: 5 – (21,16) – Yardımsever Kullanıcı 2

--

Şekil 5. hedef.txt dosyasının içeriği

6. DİKKAT EDİLMESİ GEREKENLER

1. Ödevleriniz değerlendirilirken elimizdeki farklı test dosyaları kullanılacaktır. Bu test dosyaları sizinle paylaştığımız örnek “*.txt” dosyaları ile aynı formatta olacaktır. Bu yüzden kodunuzun farklı boyutlarda veri alıp veri yazma işlemine izin verdiğinden emin olunuz.

2. Ödev C++ programlama diliyle yazılacaktır (C++ dili, C dilini desteklediğinden kodunuzun çalışıyor olması C++’a özgü hazırlanmış anlamına gelmez. C++ dilinde hazır **STL kütüphanelerini** kullanmayınız. Kullandığınız takdirde ödev değerlendirmeye alınmayacaktır. C++ diline özgü olmasına dikkat edin, nesne kullanımı, bellekte yer ayırma gibi işlemlerde dikkatli olun, malloc yerine new, free yerine delete gibi).

3. Ödevi hazırlarken doğru veri yapısını ve doğru algoritmayı kullandığınızdan emin olun!

4. Ödevleriniz Haruzem üzerinden (ue.harran.edu.tr) yüklenecektir.

Örneğin, ödevi gönderen kişi Ayşe Şeker olsun ve öğrenci numarası 123145344 olsun.

Öncelikle, ödev dosya ismi **123145344_Ayse_Seker_odev1.zip** şeklinde olacaktır. Yani göndereceğiniz ödev dosyasını, kendi öğrenci numaranız ve ad soyadınıza göre örnekte gösterildiği gibi adlandırın (dosyayı isimlendirirken Türkçe karakter kullanmayınız, boşluk bırakmayınız, ayırmayı alt çizgiler kullanarak yapınız).

6. Ödevlerinizi en geç teslim tarihi saatine (ilk sayfada) kadar göndermiş olduğunuzdan emin olunuz. **Ödev teslim tarih saatinden sonra gönderilen ödevler değerlendirilmeye alınmayacaktır!**

7. Bu ödevin, genel dönem puanına etkisi **20 puandır**. Kopya ödev, İnternette hazır kopyala yapıştır ödev teslim etmeyiniz. Bu tür durumların tespitinde ilgili öğrencinin **disiplin kuruluna sevk**i yapılacaktır. O yüzden ödevinizi kendiniz yapınız!

8. Göndereceğiniz kodun **Linux konsol (terminal)** ortamında çalışıyor olduğuna dikkat ediniz.

9. Ödevi Gönderirken sadece “*.cpp”, “*.h” dosyalarınızı gönderiniz. Ödevi gönderirken **proje halinde göndermeyiniz**. Aksi Takdirde ödev değerlendirmeye alınmayacaktır. Ayrıca ödevi gönderirken lütfen “*.exe” dosyası göndermeyiniz.

10. Ödev ile ilgili sorularınızı handangumus@harran.edu.tr ve songulakdag@harran.edu.tr mail adreslerinden hocalarımıza sorabilirsiniz.