

This report is written Turkish and English.
Bu rapor Türkçe ve İngilizce olarak yazılmıştır.

Dosya	Düzenle	Arama	Görünüm	Belge	Yardım
1 hame		arriving time		treatment time	complaint
2 Patient1		0		7	RED
3 Patient2		2		3	RED
4 Patient3		8		6	GREEN
5 Patient4		15		2	RED
6 Patient5		16		1	GREEN
7 Patient6		20		9	YELLOW
8 Patient7		21		2	GREEN
9 Patient8		22		8	RED

Dosya türü: Hiçbiri Satır: 1 Sütun: 0 ÜZRYZ

Dosya	Düzenle	Arama	Görünüm	Belge	Yardım
1 hame		arriving time		treatment time	complaint
2 Patient1		0		7	RED
3 Patient2		2		3	RED
4 Patient3		3		6	GREEN

Dosya türü: Hiçbiri Satır: 1 Sütun: 0 ÜZRYZ

```
Uçbirim - myilmaz@pardus: ~/Clouds/Dropbox/ITU/17-18 Güz/BLG233E/H -  
Dosya Düzenle Göster Uçbirim Sekmeler Yardım  
myilmaz@pardus:~/Clouds/Dropbox/ITU/17-18 Güz/BLG233E/Homeworks/Homework $ ./a.out  
1. time slice Patient1  
2. time slice Patient1  
3. time slice Patient1  
4. time slice Patient2  
5. time slice Patient2  
6. time slice Patient2  
7. time slice Patient1  
8. time slice Patient1  
9. time slice Patient1  
10. time slice Patient3  
11. time slice Patient3  
12. time slice Patient3  
13. time slice Patient1  
14. time slice Patient3  
15. time slice Patient3  
16. time slice Patient3  
17. time slice Patient4  
18. time slice Patient4  
19. time slice Patient5  
20. time slice No treatment...  
21. time slice Patient6  
22. time slice Patient6  
23. time slice Patient8  
24. time slice Patient8  
25. time slice Patient8  
26. time slice Patient7  
27. time slice Patient7  
28. time slice Patient6  
29. time slice Patient6  
30. time slice Patient6  
31. time slice Patient8  
32. time slice Patient8  
33. time slice Patient8  
34. time slice Patient6  
35. time slice Patient6  
36. time slice Patient6  
37. time slice Patient8  
38. time slice Patient8  
39. time slice Patient6  
myilmaz@pardus:~/Clouds/Dropbox/ITU/17-18 Güz/BLG233E/Homeworks/Homework $ clear
```

```
Uçbirim - myilmaz@pardus: ~/Clouds/Dropbox/ITU/17-18 Güz/BLG -  
Dosya Düzenle Göster Uçbirim Sekmeler Yardım  
myilmaz@pardus:~/Clouds/Dropbox/ITU/17-18 Güz/BLG233E/Homeworks/Homework $ ./a.out  
out  
1. time slice Patient1  
2. time slice Patient1  
3. time slice Patient1  
4. time slice Patient2  
5. time slice Patient2  
6. time slice Patient2  
7. time slice Patient3  
8. time slice Patient3  
9. time slice Patient3  
10. time slice Patient1  
11. time slice Patient1  
12. time slice Patient1  
13. time slice Patient3  
14. time slice Patient3  
15. time slice Patient3  
16. time slice Patient1  
myilmaz@pardus:~/Clouds/Dropbox/ITU/17-18 Güz/BLG233E/Homeworks/Homework $
```

-----ENGLISH REPORT-----

!!!IMPORTANT!!!

- If at the end of the file the <space>, <tab>, <newline> ... characters the last patient is read twice. So at the end of the input file, be careful not to have any characters.
- It is thought that the input file is ordered by patients according to the time of arrival in the hospital.
- This report <tab> space is written in 8 units.

~~~~ How does it process? ~~~~

### 1) Creating Queues

- The program creates a queue named "inputQ" for the data read from the file.
- Creates queues with their own names for each priority. (redQ, yellowQ, greenQ)
- OutputQ is generated for the output file.

## 2) Importing data from a file

- reads the data from the file named "patientsInfo.txt".
- First reads the header line. No action.
- Then read patient information and write to inputQ.

## 3) Time flows, program runs

- When the time starts to progress, the program starts processing.
- The program continues until the patient is unfinished.
- When each time interval passes, the following operations are performed in order.  
(In the following sections "TIME" will be used for the current time.)

```
-- if the patient in inputQ arrives at the hospital at "TIME", the patient is added
  to the queue according to priority.
-- time passes
-- treatment takes place as follows.

    --- if redQ is not empty; (During the "1" time interval;)
        ---- XXXX operations are performed.
    --- Else if yellowQ is not empty; (During the "2" time interval;)
        ---- XXXX operations are performed.
    --- Else if greenQ is not empty; (During the "3" time interval;)
        ---- XXXX operations are performed.
    --- Else
        ---- "No treatment ..." is added to outputQ.
XXXX operation
{
    ---- If the patient in inputQ arrives at the hospital in "Time",
      the patient is added to the queue according to priority.
    ---- The first patient in xxxQ is treated.
    ---- required treatment period 1 time interval is reduced
    ---- TIME increases the time interval.
    ---- patient copied to outputQ.
    ---- If the patient's treatment is over, "3) Time flows, program starts"
      step is continued.
    ---- if the treatment duration does not end, patient priority 1 is lowered.
      the patient is added to the queue according to priority.
}
```

## 4) The patient names in outputQ are printed on the screen at time intervals (with for loop).

@Author  
Muhammed YILMAZ  
150150149  
yilmazmu15@itu.edu.tr

-----TÜRKÇE RAPOR-----

!!!ÖNEMLİ!!!

- Eğer dosya sonunda <boşluk> , <tab> , <yeni satır> ... karakterlerinden var ise son hasta iki defa okunur. Bu yüzden input dosyasının sonunda öncelikten sonra hiç bir karakter olmamasına dikkat ediniz.

- İnpıt dosyasınının, hastaların hastaneye geliş zamanına göre sıralı olarak verildiği düşünülmüştür.
- Bu rapor <tab> boşluğu 8 birim şeklinde yazılmıştır.

~~~~ Nasıl çalışır? ~~~~~

1) Queue'ları oluşturma

- Program dosyadan okunan veriler için "inputQ" adında bir queue oluşturur.
- Her öncelik için kendi isimlerinde queue'lar oluşturur. (redQ, yellowQ, greenQ)
- Çıktı dosyası için "outputQ" olulturulur.

2) Dosyadan verileri alma

- "patientsInfo.txt" isimli dosyadan verileri okur.
- Öncelikle başlık satırını okur. İşlem yapmaz.
- Ardından hasta bilgilerini okur ve inputQ'ya yazar.

3) Zaman akar, program çalışır

- Zaman ilerlemeye başlayınca program da işlemeye başlar.
- Tedavisi bitmemiş hasta kalana kadar program devam eder.
- Her 1 zaman aralığı geçerken sırasıyla aşağıdaki işlemler gerçekleşir.
(Bundan sonraki kısımlarda şimdiki zaman için "TIME" kullanılacaktır.)

- inputQ içerisindeki hasta hastaneye "TIME"da geldiyse, hasta önceliğine göre ilgili queue'ya eklenir.
- zaman ilerler
- tedavi işlemi aşağıdaki şekilde gerçekleşir.

```
--- if redQ boş değilse;          ("1" zaman aralığı boyunca;)
    ---- XXXX işlemleri yapılır.
--- Else if yellowQ boş değilse;  ("2" zaman aralığı boyunca;)
    ---- XXXX işlemleri yapılır.
--- Else if greenQ boş değilse;   ("3" zaman aralığı boyunca;)
    ---- XXXX işlemleri yapılır.
--- Else
    ---- outputQ'ya "No treatment..." eklenir.
XXXX işlemi
{
    ---- inputQ içerisindeki hasta hastaneye "Time"da geldiyse, hasta
        önceliğine göre ilgili queue'ya eklenir.
    ---- xxxQ'daki ilk hasta tedavi edilir.
    ---- gereken tedavi süreli 1 zaman aralığı azalır
    ---- TIME bir zaman aralığı artar.
    ---- hasta outputQ'ya kopyalanır.
    ---- Eğer hastanın tedavisi bittiyse "3) Zaman akar, program çalışır"
        adımından devam edilir.
    ---- tedavi süresi bitmezse hasta önceliği 1 düşürülür. 1 alt öncelik
        queue'sunun sonuna eklenir.
}
```

4) outputQ içerisindeki hasta isimleri zaman aralıklarıyla (for içinde) ekrana yazdırılır.

@Hazırlayan
Muhammed YILMAZ
150150149
yilmazmu15@itu.edu.tr