

## Senaryo:

Bir hastane acil servisi, gelen hastaları tedavi önceliğine göre sıralamak istiyor. Her hasta için şu bilgiler mevcut:

- Hasta ID (key):** Her hastaya özel bir kimlik numarası.
- Öncelik Seviyesi (value):** Hastanın durumunun aciliyetini temsil eder (1: en acil, 10: düşük öncelik).
- Tedavi Süresi:** Her hastanın tedavi süresi (dakika cinsinden).

Hastane şu işlemleri yapmak istiyor:

- Hastaların öncelik seviyesine göre sıralanması:** Daha düşük öncelik seviyesine sahip hastalar daha önce tedavi edilir.
- Tedavi Süresine Göre Zaman Yönetimi:** Her hasta tedavi edilirken, toplam tedavi süresinin gün sonunda 420 dakikayı (7 saat) aşmamasını sağlamak.
- Dinamik Ekleme ve Silme İşlemleri:** Gün içinde gelen yeni hastalar kuyruk yapısına eklenir. Tedavi edilen hastalar ise kuyruktan çıkarılır.

## Hasta Listesi:

HastaID	Öncelik Seviyesi	Tedavi Süresi(dk)
101	5	30
102	3	40
103	8	20
104	1	60
105	7	15
106	2	50
107	4	45
108	6	25
109	3	35
110	2	30
111	8	10

## Görevler:

- Min-Heap Oluşturma:** Başlangıç hasta listesinden bir Min-Heap yapısı oluşturun. Öncelik seviyesi sıralama kriteri olarak kullanılacaktır.
- Dinamik Hasta Ekleme:** Gün içinde gelen yeni hastaları heap'e ekleyin ve heap'i düzenleyin.
- Tedavi Simülasyonu:** Gün sonunda 7 saatlik toplam tedavi süresini aşmadan, hastaları sırasıyla tedavi edin ve hangi hastaların tedavi edildiğini listeleyin.
- Tedavi Edilemeyen Hastalar:** Gün sonunda tedavi edilemeyen hastaları ayrı bir liste olarak yazdırın.
- Kod Yapısı:** Kodunuzun temiz, okunabilir ve açıklama satırlarıyla desteklenmiş olması gerekmektedir.

## İstenen Belgeler:

Kaynak Kod Dosyası (.cpp VEYA .py)

Derleme ve Çalıştırma Talimatları (README.md)

Test Giriş/Çıkış Dosyaları (input.txt ve output.txt)

Kodun doğru çalıştığını gösteren ekran çıktıları

Kısa bir video ile çalışır program anlatımı (Max 1 dk)