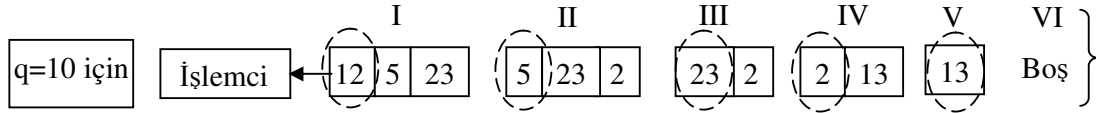


Numara :  
İsim :

**EGE ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**204 VERİ YAPILARI (3+1)**  
**2008-2009 GÜZ YARIYILI**  
**VİZE (120 dakika)**

**85 Puan +** (Proje1 ve Proje 2, toplam 15 puan üzerinden değerlendirilmektedir)

1) Round robin iş zamanlaması algoritmasında, işlerin boyutları (büyüklüğü) değişken olabilmektedir. Quantum (q) adı verilen belli bir boyuttan büyük işlerin q boyutu kadarlık kısmı tamamlanıp, kalanı kuyruğun sonuna tekrar eklenmektedir. q boyutundan küçük (veya eşit) işler, boyutları kadar sürede tamamlanıp kuyruktan tamamen silinmektedir:



Bu işlemi gerçekleştiren kuyruk sınıfını, enqueue (ekle) ve dequeue (sil) metodlarını yazınız. Dequeue içerisinde, tamamı bitirilemeyen işlerin kuyruğa belirtilen şekilde eklenmesini de gerçekleştiriniz. Tüm işlemlerin, yaklaşık aynı anda geldiklerini varsayabilirsiniz. Sil metodu içinde, etkin işin ve (o ana kadar tamamlanan işler için) ortalama tamamlanma süresini de hesaplatıp yazdırınız. Kuyruk altyapısında Vector kullanınız. (20 puan)

2) Sırt çantası probleminde, ağırlıkları (boyutları) ve değerleri olan eşyalar (parçalar) bulunmaktadır. **a) Kapasitesi (alabileceği toplam ağırlık) verilen** bir sırt çantasına, özellikleri verilen parçalardan en fazla ne kadar **toplam değerde** eşya yerleştirilebileceğini bulan özyineli (recursive) bir metod/fonksiyon (Java/C) yazınız (10 puan). **b) Yönteminizi anlatınız.** (10 puan) Parçaların özelliklerinin bir dizide tutulduğunu ve i. parçanın boyutuna parca[i].boyut, değerine parca[i].deger şeklinde ulaşabileceğinizi varsayabilirsiniz:

	0	1	2	3	4
Parça	A	B	C	D	E
Boyut	3	4	7	8	9
Deger	4	5	10	11	13



**Çanta kapasitesi =17 için, bu örnekte 24** bulunmalıdır: { D, E }, { A, C, C } örnekleri olası en yüksek değeri vermektedir. { A,A,B,C } kapasiteyi doldurmakla beraber, Çantanın toplam değerini 23 yapabilmektedir.

Metodunuz sadece yukarıdaki örneği değil, tüm verilecek parça dizisi ve kapasitelerdeki Sırt Çantası problemlerinin çözümünü yapabilmelidir. Her bir parça tipinden sınırsız sayıda bulunduğunu varsayabilirsiniz. Bu sorunun çözümünde, Yığıt, Kuyruk veya Ağaç gibi bir veri yapısı kullanmayınız!

Puanlamada çözümün doğruluğu ve kodun gereksiz yere uzatılmaması da dikkate alınacaktır.

3) Verilen bir ikili ağacın (binary tree), N. düzeyindeki yaprakların sayısını bulduran, bulunan düğümleri (referanslarını) yığta atan etkin fonksiyonu yazınız. Hazır yığıt fonksiyonlarını kullanabilirsiniz. (15 puan) **Not:** Tüm ağaç sınıfını ve metodlarını yazmanıza gerek yoktur.

#### 4) Sınavın Test Bölümü

Arka sayfadaki test sınavı, her biri 3'er puan olan 10 sorudan oluşmaktadır ve toplam 30 puan değerindedir. Yanlışlar doğruları götürmemektedir.