

CMPE 242

Programlama Ödevi 1

Bu ödev 4 Nisan Pazar 23: 55'e kadar teslim edilecektir.

HW ile ilgili sorularınızı sorabilirsiniz. Lütfen şu seçeneklerden birini kullanın:

- Moodle Ev Ödevi **Soru Forumu** : HW Soru-Cevap (Soru-Cevap) Forumu
Moodle her zaman müsaittir. Kurs Moodle sayfasındaki "Forum" bağlantısını kullanın.
- Ev Ödevi **BAŞVURU SAATLERİ** : İki Soru-Cevap **BAŞVURU SAATI** olacaktır.
takip eden günler:
 - CMPE242-HW1-OfficeSaat1: 22 Mart, 06: 00-07: 00, Yakınlaştırma Kimliği: 952 6936 0016
 - CMPE242-HW1-OfficeHour2: 26 Mart, 06: 00-07: 00, Yakınlaştırma Kimliği: 928 5510 3025

Not: Lütfen katılmadan önce HW belgesini iyice okuduğunuzdan emin olun.

BÖLÜM I

Aşama 1

İlk kısım için, bağlantılı bir liste kullanarak Genel Yığın ADT'sini uygulamalısınız. Mecbursun kendi bağlantılı liste uygulamamızı sıfırdan uygulayın, yerleşik Java'yı kullanamazsınız bağlantılı liste sınıfı. Yığın uygulamamızı **myStack** olarak **adlandırın** ve sınıfın uygulaması gerekir aşağıdaki işlemler:

boolean isEmpty ()	Yalnızca yığın boşsa "true" değerini döndürür.
int size ()	Yığındaki öğe sayısını döndürür.
void push (Öge)	Bir öğeyi yığının en üstüne iter.
Öge pop ()	En üstteki öğeyi yığının üstünden kaldırır ve döndürür.
Öge gözetleme ()	Yığına en son eklenen öğeyi, kaldırarak.
Yazdır()	Yığının öğelerini yukarıdan aşağıya doğru yazdırır.

Not 1: Tüm yöntemleri ve tüm istisnaları da test ettiğinizden emin olun.

Not 2: Sınıfı uygulamadan önce jenerik kavramını anladığınızdan emin olun.

Sayfa 2

Adım 2:

Bu adımda, oluşturduğunuz **myStack** sınıfını kullanacak bir **VaccineStock** uygulaması **yazacaksınız**. COVID-19 aşı stoğunun yönetimi için yukarıda verilmiştir. (*Not: normalde bir yığın kullanmazsınız Bu amaç için gerçek hayatta, ancak yığınların kullanımını deneyimlemenizi istiyoruz.*)

Önce **VaccineStock** adlı bir sınıf oluşturacaksınız . Bu sınıfın üç özel veri üyesi olacak ve üç işlev:

seri numarası	Aşı seri numarasını tutan bir tam sayı.
ülke adı	Aşının yapıldığı ülkenin adını tutan bir dize imal edilmiştir.
numberOfVaccines	Üretilen aşı sayısını tutan bir tam sayı.
void popItem ()	Ögeyi yığından çıkarır ve görüntüler.
void pushItem ()	ögeyi yığının üzerine iter
int eylem ()	İşlem menüsünü görüntüler ve kullanıcının seçimine geri döner

Bu programa sonsuz döngüde **ADD / DELETE / EXIT** komutunu girmesi istenmelidir . Kullanıcı **EKLE** seçeneğine girdiğinde, program **myStack** envantere yeni bir öğe ekler ; içinde **SİL** girişi **myStack** envanterinden bir öğeyi kaldırır . Döngü, kullanıcı **EXIT** yazana kadar devam etmelidir . Bir öğe eklerken, program kullanıcıdan üç öğe için ihtiyaç duyduğu bilgileri istemelidir.

VaccineStock sınıfının veri üyelerini seçin ve yığına yeni bir öğe ekleyin. Bir yığından öğe varsa, program **VaccineStock** nesnesindeki tüm bilgileri görüntülemelidir. yığından fırlatıldı. Program sona erdiğinde, kalan tüm öğeleri açmalıdır Yığının ve verilerini görüntüleyin. Aşağıda örnek çıktı biçimini görebilirsiniz:

3. Sayfa

```
$ java Aşısı
COMMAND girilsin mi?
EKLE
ÖGE VERİLERİ girilsin mi?
191
ÇİN
15.000
COMMAND girilsin mi?
EKLE
ÖGE VERİLERİ girilsin mi?
192
TÜRKİYE
8000
COMMAND girilsin mi?
SİL
192, TÜRKİYE, 8000
COMMAND girilsin mi?
ÇIKIŞ
191, ÇİN, 15000
```

4. sayfa

BÖLÜM II

Bölüm II için, yeniden boyutlandırma dizisi yaklaşımını kullanarak Queue ADT'yi uygulamalısınız. Malısın dizi boyutu 4 ile başlayın ve dizinin boyutu, aşağıdaki durumlarda artırılabilir (iki katına çıkarılabilir) veya azaltılabilir (yanya indirilebilir) sınıfta açıklandığı gibi gerekli.

Aşama 1:

Genel bir Queue sınıfı yazın. Sınıfınızı **myQueue** olarak **adlandırın** . **MyQueue** sınıfı gerektiği aşağıdaki işlemleri uygulayın. Kendi uygulamamızı uygulamamız gerektiğini unutmayın.

boolean isEmpty ()	Yalnızca kuyruk boşsa "true" değerini döndürür.
boolean isFull ()	Yalnızca ve ancak kuyruk doluyrsa "true" değerini döndürür.
int size ()	Kuyruktaki öğelerin sayısını döndürür.
void enqueue ()	Ögeyi sıranın arkasına ekler.
Sıradan çıkan öğe ()	Sıranın önünden bir öğeyi siler.
Öge gözetleme ()	Sıradaki ön öğeyi silmeden döndürür.
int eylem ()	İşlem menüsünü görüntüler ve kullanıcının seçimine geri döner.
void print ()	Kuyruğun öğelerini önden arkaya yazdırır.

Not 1: Tüm yöntemleri ve tüm istisnaları da test ettiğinizden emin olun.

Not 2: Sınıfı uygulamadan önce jenerik kavramını anladığınızdan emin olun.

Adım 2:

Simüle etmek için yukarıda oluşturduğunuz **myQueue** sınıfını kullanacak bir **Aşı** uygulaması yazın.

Sağlık ocağında COVID-19 aşısı bekleyen öğrenci hattının yönetimi Üniversite.

İşte varsayımlar:

- Sağlık merkezinin günlük aşılama kapasitesi X'tir (yani, sağlık ocağında yalnızca X kişiyi aşılayabilir. verilen gün).
- Öğrenciler, bir adı ve boyutu olan öncelikli risk gruplarına girerler. Örneğin Boyut 40, Grup2 boyut 30 ile Grup1...
- Grup büyüklüğünün sağlık merkezinden fazla olmaması garantilidir (yani varsayabilirsiniz) günlük aşı kapasitesi (X).
- Sağlık ocağı öncelikle en yüksek risk önceliklerine sahip grupları aşılamalıdır.
- Sağlık ocağı günlük aşılama kapasitesini doldurmalıdır.

5.Sayfa

Program girdi dosyası COVID19.txt'dir. Dosyanın ilk satırı günlük kapasiteyi içerir Sağlık ocağı sayısı. Aşağıdaki satırlar, bir ad ve bir numara içerir.

grupun adı ve gruptaki öğrenci sayısı. Gruplar azaltıyor öncelik sırası (yani ilk grup en yüksek önceliğe sahiptir).

Giriş:

```
50
Grup1 30
Grup2 10
Grup3 50
Grup4 20
```

Çıktı aşağıdaki biçime sahip olmalıdır: Her gün için, grup adlarının bir listesi ve o gün aşılanan boyut. Örnek çıktı formatına bakın. Son olarak, toplam sayıyı yazdırın aşılanan öğrencilerin yüzdesi.

Çıktı:

```
1. Gün: 1. Grup 30 2. Grup 10
2. Gün: 3. Grup 50
3. Gün: 4. Grup 20
Toplam Öğrenci: 110
Toplam Gün: 3
```