

Github Classroom ile Hafta 7 - Vize

Uygulama için sizinle paylaşılan link ile oluşturulan Github kod deposu (repository) içerisine sadece sizden istenilen ve aşağıda belirtilen formatta dosyayı yükleyiniz. Yükleyeceğiniz dosya isimlerinde ve yazacağınız kodda fonksiyon isimleri, giriş argümanları ve çıkış tipleri belirtilen şekilde olmalıdır. Sistemin yazdığınız kodları notlandırabilmesi için yazdığınız kodun çalışıyor olması gerekmektedir.

Yüklenecek dosya adı: **hafta7_vize.c** ve **hafta7_vize.docx**

Yazmanız gereken açıklamalar: (Kod ve Yorum dosyasına)

hafta7_vize.docx: Bu dosya içerisine Aykırı değer ne olduğunu ve ne işimize yaradığını kendi cümlelerinizle anlatınız. Yazdığınız kodun içerisinde kullandığınız işaretçi kullanımından bahsediniz ne için kullandığınızı yazınız. Yazdığınız kodun algoritma ve uygulama bakımından nerelerde kullanılabileceğini, güçlü ve zayıf yönlerini düşünerek kendi cümlelerinizle yazınız.

Not: Sizlerden kitap tanımı veya uzman görüşü beklemiyoruz. Özellikle kendi yazdığınız kod ve algoritmaların nasıl çalıştığını, nerelerde kullanılabileceğinizi ve nasıl geliştirebileceğinizi düşünerek algoritma ve programlama yeteneğinizi artırmak birincil hedef olarak belirlenmiştir. Bu sebeple nerede neyi hangi amaçla yaptığınızı belirtiniz. Gerekli durumlarda word dosyası içerisinde kodunuzdan ekran görüntüleri alarak açıklayabilirsiniz.

Hafta7_vize.c: Bu dosyada algoritmanızı yazarak unitTest.c dosyasında çalıştırınız. Notlandırmada verilen giriş değerleri dışında farklı değerlerle test edilebilir olduğunu unutmayın. Yazdığınız kod parçalarının açıklamalarını yorum satırlarında açıklayınız.

Puanlama Tablosu (Grading Rubric):

Kriterler	Puan
İşaretçilerin doğru kullanımı	20
Filtreleme işleminin ve fonksiyonların gerçekleşmesi	20
Kod okunabilirliği ve dokümantasyonu	5
Düzenli programlama pratiklerine uygunluk (kod düzeni vb.)	5
Aykırı değer bulma ve filtreleme algoritmalarının açıklaması	10
Kullanılan işaretçi aritmetiklerinin açıklaması	15
Kodların c dosyası içerisinde açıklamaları	15
Yazdığınız kodun güçlü ve zayıf yönlerini yorumlama	10
(Ekstra) Aynı değerlerin işaretçi dizisiyle ayıklanması	10
(Ekstra) İşaretçi dizisi ayıklama işleminin yorumlanması	10

***Not:** Sınavdan en fazla 100 puan alınabilir.

Oluşturmanız İstenilen Algoritma ve Kod Hakkında

Hafta7.c dosyası içerisinde

```
void hafta7_fonk(int dizi[], int elemanSayisi, int yuzdelikDeger, double *Q1ptr, double *Q3ptr, double *IQRptr,
double *OrtPtr, double *ptr_esikDegeri_Alt, double *ptr_esikDegeri_Ust,
int *islemeAlinanDegerlerFlag, double *ptr_IADT, int *ptrArray[]);
```

Fonksiyon prototipi uyarınca kod yazacaksınız. Bu fonksiyon içerisinde sizden istenilen adımlar:

ADIM-1: Aykırı değer hesaplaması yapmanız. Önceki haftalarda yazdığınız kodları kullanarak burada Q1, Q3, ve IQR değerlerinin hesaplamasını ve atamasını yapabilirsiniz.

“islemeAlinanDegerlerFlag”: değişkeni adımlarda işleme almak istemediğiniz dizi değerlerini belirleme (eklemek veya çıkartmak) için kullanılır. İşleme almak istiyorsanız 1 istemiyorsanız 0 yaparak dizinin o elemanını yapılacak işlemlere dahil edebilirsiniz. Bu dizi giriş olarak gelen “dizi” değişkeni ile aynı eleman sayısına sahiptir (“elemanSayisi”).

ADIM-2: Aykırı değerlerin sonraki yapılacak işlemlerde işleme alınmamasını sağlayın.

ADIM-3: Aykırı olmayan değerlerin ortalamasını hesaplayınız. “OrtPtr” kullanarak bu değeri işaretçinin gösterdiği adrese atayınız. Fonksiyonda size argüman olarak verilen “yuzdelikDeger” kullanarak bu ortalama değer ve yüzdelik değeri kullanarak alt ve üst eşik değerleri belirleyin.

$$\begin{aligned} \text{YüzdelikAralık} &= \mu * \text{YüzdelikDeğer} \\ \text{Alt Eşik Değeri} &= \mu - \text{YüzdelikAralık} \\ \text{Üst Eşik Değeri} &= \mu + \text{YüzdelikAralık} \end{aligned}$$

μ değeri ortalama değeri ifade etmektedir. Bu eşik değerlerini, “ptr_esikDegeri_Alt” ve “ptr_esikDegeri_Ust” işaretçilerini kullanarak gösterdiği adreslere atayınız.

ADIM-4: Önceki adımda belirlediğiniz eşik değerleri arasındaki dizi değerlerini işleme almak için işleme alınan değerler dizisini (“islemeAlinanDegerlerFlag”) güncelleyiniz.

ADIM-5: Bu aşamada sıralanmış dizi içerisindeki eşik değerleri arasında yer alan dizi değerlerini toplayarak “ptr_IADT” işaretçisi ile hafızada gösterilen adrese bu toplam değeri yazınız.

ADIM-6: Bu adımda “ptrArray” isimli işaretçi dizisi kullanılacaktır. İşaretçi dizisinin bütün elemanları başlangıç olarak NULL değerine atanmıştır. Burada önceki adımlarda elde ettiğiniz aykırı olmayan ve eşik değerleri arasında yer alan dizi değerlerinden aynı değere sahip olmayanları (duplicate) yani bulduğunuz ilk eşsiz değerleri (unique) bularak bu değerleri barındıran dizi adreslerini “ptrArray” dizisinin aynı indisteki bölgesine atama işlemini gerçekleştiriniz. Örneğin dizinin 4. ve 5. indislerinde eğer 19,19 şeklinde aynı değer var ise işaretçi dizisindeki ptrArray[4] değeri dizi[4] indisinin değerini barındırmalı ptrArray[5] ise NULL değeri olarak kalmalıdır.

Burada belirtilen adımlarda yapılması zorunlu olmayıp doğru olduğu sürece istediğiniz sıra ve yöntemle işlemlerinizi gerçekleştirebilirsiniz. Yalnızca kodu ve algoritmaları kendiniz yazınız. Göndereceğiniz kodlar intihal programından geçirilerek değerlendirilecektir. Bu sebeple kod ve kod satırlarında yazacağınız yorumları kendiniz yazınız.

Not: unitTest.c dosyası, **hafta7_vize.c** dosyasında yazdığınız kodu puanlama tablosu uyarınca 50 puan üzerinden değerlendirecektir. Diğer puanlar dersin yürütücüsü tarafından inceleme sonucunda verilecektir.

Yükleyeceğiniz dosya ile ilgili önemli noktalar:

- Aksi belirtilmediği takdirde istenilen dosyada ana fonksiyon (*main* fonksiyonu) yazmayınız. (Aksi takdirde sistem derleme işleminde hata oluşur ve puan alamazsınız.)
- Dosyada fonksiyon tanımları (dosya içerisinde burada prototipi verilen fonksiyonların kodları yer almalıdır.)
- “*stdio.h*” kütüphanesi ve test için gerekli kütüphaneler *unitTest.c* dosyasında tanımlanmıştır. Kullanacağınız başka diğer kütüphaneler var ise yükleyeceğiniz dosya içerisinde en başta tanımlayınız.
- Fonksiyon adı, kullanılan argüman ve geri-dönüş tipi birebir aynı olmalıdır. Kod yazarken fonksiyon bloğunun içerisinde istediğiniz değişkenleri istediğiniz isimde isimlendirip kullanabilirsiniz. (Hata almamak için Türkçe karakter kullanmamaya özen gösteriniz.)
- Sadece sizden istenilen dosya içerisinde algoritmanızı yazınız. Özellikle sistemi yanıltmaya yönelik girişimlerde bulunmayınız. Sistemi yanıltmaya yönelik çalışmalar olması durumunda gerekli işlemler yapılacaktır. (Kasıtlı olarak sistem dosyalarının değiştirilmesi vb.)
- Yüklediğiniz kodlar benzerlik tarama sisteminden geçirilecektir. Lütfen kendi yazdığınız kodları sisteme yükleyiniz.

Vize Dosyaları için Depo Yükleme Son Tarihi ve Saati: 07.04.2023 23:59

Uygulama veya Sistem ile ilgili bir hata görürseniz lütfen dersin hocası ile iletişime geçiniz.