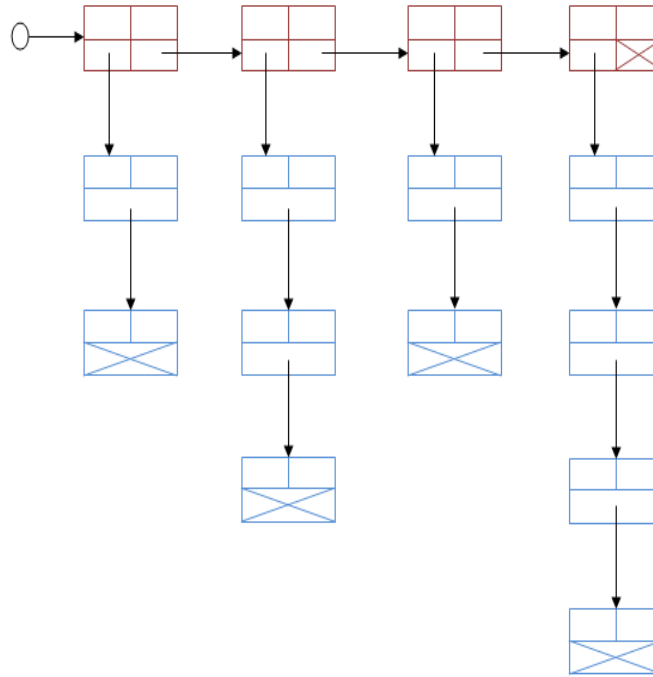


Ankara Üniversitesi
Bilgisayar Mühendisliği
BLM2067/BLM267 Lab4

Bu uygulama kapsamında size bir main.c ve function.h dosyası verdik. main.c dosyası sadece ana fonksiyonunuz olan int main() fonksiyonunu içerirken function.h dosyası bu uygulama kapsamında kullanacağınız diğer bütün fonksiyonların tanımlarından oluşacak. Sizden beklenen main.c dosyasının beklenen çıktıyı oluşturacak şekilde hatasız olarak çalışması için function.h dosyasının içeriğini doldurmanız. Öncelikle main.c dosyasını dikkatle inceleyiniz. Burada kullanıcıdan değer alma işlemleri ile ilgili fonksiyonları çağırma işlemlerini biz sizler için tanımladık. Fonksiyonların ayrı bir dosyada tanımlanması işlemi için YardimciDokuman.pdf dokümanını inceleyebilirsiniz.

function.h dosyasının içeriğini tamamladığınızda bir öğretim görevlisinin verdiği derslerin bilgileri ile bu dersi alan öğrencilerin listesini tutan bir C programı elde edeceksiniz. Bu öğretim görevlisi 4 farklı sınıfa ders vermektedir. Her bir sınıftaki öğrenci sayıları birbirinden farklı olabilir. Dersleri ve öğrencileri tutmak için iki farklı bağlı liste yapısı bulunmaktadır. Aşağıda bu listelere ait düğüm yapıları verilmiştir. Sınıfa ait yapıda (*nodeClass*), kaçınıcı sınıf dersi olduğu bilgisi (*classID*) ve sınıftaki öğrencilerin arasnavlarının ortalaması (*classMidtermAverage*) tutulmaktadır. Bunların yanında bir sonraki sınıfa gösteren bir işaretçi ile öğrencilerin bilgilerini tanımlamak için kullanılan yapıya (*nodeStudent*) ait bir düğümü gösteren bir işaretçi bulunmaktadır. *nodeStudent* olarak isimlendirilen yapıda, öğrencinin numarası, arasnav notu ve aynı sınıftaki bir sonraki öğrenciyi gösteren bir işaretçi bulunmaktadır. Bu iki yapının birbiri ile olan ilişkisi aşağıdaki şekilde verilmiştir. Kırmızı düğümler *nodeClass* yapısına, mavi düğümler *nodeStudent* yapısına ait düğümlerdir.



Program kullanıcıdan öğrenci numarası ve arasnav notunu girdi olarak almaktadır. Öğrenci numarası 66 ile başlayan öğrenciler 1. sınıfta, 77 ile 2. sınıfta, 88 ile başlayan öğrenciler 3. sınıfta ve 99 ile başlayan öğrenciler 4. sınıftadır. Öğrenciler bağlı listede sıralı olarak bulunmalıdır. Sıralama işlemi arasnav notuna göre azalan sırada yapılacaktır. Eğer notlar aynı ise numarası küçük olan listede önde bulunmalıdır. Ekleme işlemi yapılırken doğru konuma eklenerek sıralı liste özelliği korunmalıdır. Bütün öğrenciler listeye

eklendikten sonra her sınıfın ara sınav ortalaması hesaplanacak ve ilgili sınıfa ait *nodeClass* yapısından üretilen düğümün *classMidtermAverage* değişkeninde tutulacaktır.

Programınız *printAll* fonksiyonu ile size verilen çıktı dosyasındaki formata uygun olarak, her sınıfın, kaçınıcı sınıf olduğunu, arasınay ortalamasını ve ilgili sınıftaki öğrencilerin numaraları ile arasınay notlarını sıralı bir şekilde ekrana yazdıracaktır.

```
struct nodeClass
```

```
{  
    int classID;  
    double classMidtermAverage;  
    struct nodeClass *next;  
    struct nodeStudent *studentPtr;  
};
```

```
struct nodeStudent
```

```
{  
    int studentID;  
    int midterm;  
    struct nodeStudent *next;  
};
```