



INF 212 Algorithms and Programming II

**2020-2021 Spring
Electronic Engineering**

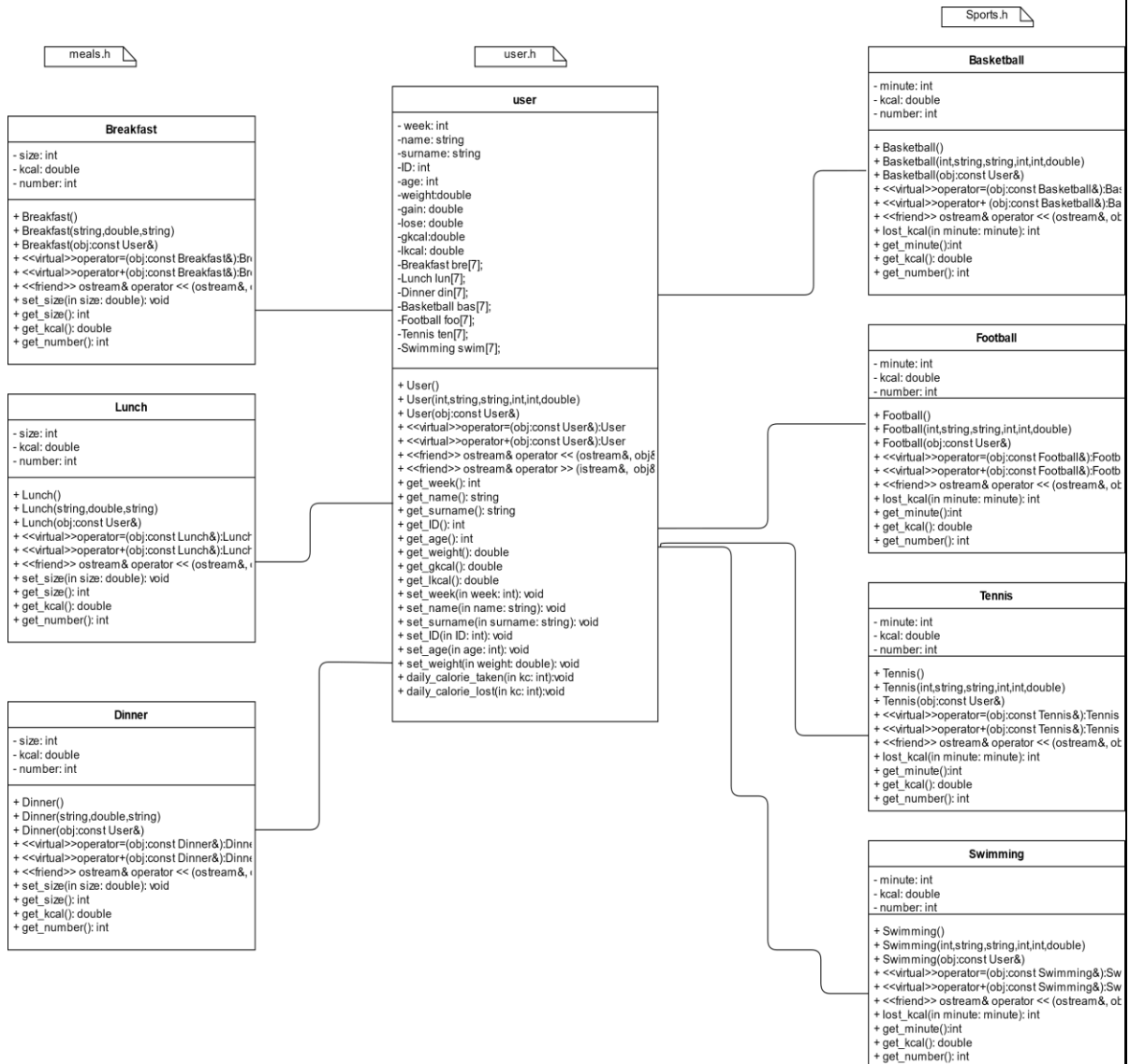
Project 3

Calorie Tracker

School ID	Name	Surname
1901022022	Selimhan	Aygün

UML CLASS DIAGRAM

Append here the UML class diagram you have created as an SVG image.
SVG formatında oluşturduğunuz UML diyagramını bu bölüme ekleyiniz.



MANUAL for USER MODE

Append here the screenshots and instructions for the user of the program.

Programı kullanacak kişi için ekran görüntülerini ve talimatları bu bölüme ekleyiniz.

```
1. Developer Mode
2. User Mode
Enter your choice:
```

Sekil-1

Programımızı başlattığımız zaman Sekil-1’de gözüktüğü üzere bizi 2 seçenek karşılıyor. User kısmı için 2 yazıp giriş yapıyoruz.

```
1. Login
2. Register
Enter your choice:
```

Sekil-2

Ardından kullanıcıdan giriş yapması veya kayıt olmasını isteyen Sekil-2’deki ekran çıkıyor. Bu kısımda dosya sistemi olmadığı için her girişimizde ilk önce kayıt olmamız gerekmektedir. Programda yanlış girişler kabul edilmemektedir.

```
Enter user's name: selim
Enter user's surname: aygun
Enter user's age: 21
Enter user's ID number: 123
Enter the user's weight: 70
Enter the week number: 1
Press a button to continue...
```

Sekil-3

Sekil-3’teki kısımda sırasıyla kullanıcı bilgilerimizi giriyoruz. Giriş kısmında ID sorulacağı için ID’yi hatırlayabileceğimiz bir şey koymamız önemli. Bu kısımda kullanıcı maksimum 7 kayıt oluşturup istediği hesapla giriş yapabilir. İşlemimiz bittikten sonra bir tuşa basıp devam ediyoruz. Sekil-2’deki ekrandan giriş kısmını seçtikten sonra Sekil-4’deki gibi ID ile giriş yapıyoruz.

```
1. Login
2. Register
Enter your choice: 1
Please enter your id: 123
```

Sekil-4

```
1. Calculate calories after sport
2. Calculate calories after meal
3. Show how many meals has eaten
4. Show how many sport has done
5. Show calories gained and lost weekly
6. Exit
Enter your choice:
```

Sekil-5

Girişimizi yaptıktan sonra Sekil-5’teki kullanıcı menüsü ekranı karşımıza çıkıyor. Bu kısımda her yemekten ve her spordan sonraki kalori değişimini görebiliyoruz. Ayrıca program yemek ve spor takibini yapabiliyor, haftalık olarak alınan ve verilen kaloriyi hesaplayıp net kaloriyi de gösterebiliyor.

```
For which day do you want to enter value? (please enter only small letters)
monday
1. Basketball
2. Football
3. Tennis
4. Swimming

Which sport is user doing? :4
How many minute is user doing this sport?20

140 kcal has been given

Press a button to continue...
```

Sekil-6

Sekil-5'te ilk seçeneği seçtiğimiz zaman Sekil-6'daki görüntü karşımıza çıkıyor. Program ilk önce sporun yapıldığı günü soruyor. İngilizce ve küçük harflerle spor yapılacak gün bilgisini girdikten sonra karşımıza hangi sporun yapıldığını soran 4 seçenek geliyor. Seçeneklerden birini seçtikten sonra bu sporun kaç dakika yapıldığını soran bir ekran geliyor ve bu kısımda da bir tamsayı girildikten sonra program, spora göre ve o gün için kaç kalori yakıldığını gösteriyor. Bu kısımda 140 kilo kalori yakılmış

```
For which day do you want to enter value? (please enter only small letters)
monday
1. Breakfast
2. Lunch
3. Dinner

Which meal is user eating? :2
1. Small
2. Medium
3. Large

Which size of a meal does the user eat?: 3

800 kcal has been taken

Press a button to continue...
```

Sekil-7

Sekil-6'nın sonunda bir tuşa basınca tekrar kullanıcı menüsüne (Sekil-5) dönülüyor. Ardından ikinci seçenek olan yemeğe göre kalori takibi kısmına girildikten sonra Sekil-7'deki ekran karşımıza çıkıyor. Yine aynı şekilde gün bilgisi giriliyor ve yenilen öğün seçiliyor. Küçük/Orta/Büyük şeklindeki öğünün büyüklüğü de seçildikten sonra program ekrana o gün için kaç kalori alındığını basıyor. Bu kısımda 800 kilo kalori alınmış

```
Number of Breakfast Per Week: 0
Number of Lunch Per Week: 1
Number of Dinner Per Week: 0
Press a button to continue...
```

Sekil-8

Sekil-5'teki 3. Seçenekteki haftalık yenilen yemek takibi seçildiğinde ise karşımıza hangi yemekten kaç defa yenildiğini gösteren bir ekran (Sekil-8) çıkıyor. Bu kısımda ben bir defa Öğle yemeği girişi yaptığım için sadece onu 1 olarak gösteriyor.

```
Number of Basketball Per Week: 0
Number of Football Per Week: 0
Number of Tennis Per Week: 0
Number of Swimming Per Week: 1
Press a button to continue...
```

Sekil-9

Sekil-5'teki 4. Seçenek olan haftalık yapılan spor takibi seçildiğinde ise karşımıza hangi spordan kaç defa yapıldığını gösteren bir ekran (Sekil-9) çıkıyor. Bu kısımda ben bir defa yüzme etkinliği girişi yaptığım için onu 1 olarak gösteriyor.

```
Net kcal of monday :660
Net kcal of tuesday :0
Net kcal of wednesday :0
Net kcal of thursday :0
Net kcal of friday :0
Net kcal of saturday :0
Net kcal of sunday :0
The amount of calories consumed by the user: 800
The amount of calories given by the user: 140
Press a button to continue...
```

Sekil-10

Sekil-5'teki kısımdan 5. Seçenek olan “haftalık kazanılan ve verilen kaloriyi göster” seçeneği seçildiğinde ise karşımıza Sekil-10'daki ekran çıkıyor. Bu kısımda program , haftanın her günündeki net kaloriyi hesaplıyor. Ayrıca bir hafta içinde toplam alınan ve toplam verilen kaloriyi gösteriyor.

MANUAL for DEVELOPER MODE

Append the screenshots and explanations for the outputs produced when the class member functions and variables are tested. Programı oluşturan sınıfların içerisindeki fonksiyonları ve operatörleri test amaçlı çalıştırdığınızda çıkan ekran görüntülerini ve açıklamaları bölüme ekleyiniz.

```
--Developer Menu--  
  
*****<User Class Tests>*****  
--User Info--  
ID: 0  
Name: NULL  
Surname: NULL  
Age: 0  
Week: 0  
Weight: 0  
-Default constructor has been tested-
```

Sekil-11'de Sekil-1'deki Developer Menu seçildikten sonraki ekranı görüyoruz. İlk kısımda User sınıfı test ediliyor.

Bu kısımda user sınıfından obje (us) oluşturuluyor ve default constructor test ediliyor.

Sekil-11

```
ifstream/ofstream operator is being tested  
Name: selim  
Surname: aygun  
Age: 21  
ID: 123  
Week: 1  
Weight: 70  
--User Info--  
ID: 123  
Name: selim  
Surname: aygun  
Age: 21  
Week: 1  
Weight: 70  
-istream/ostream has been tested-
```

Sekil-12

Sekil-11'in devamında yani Sekil-12'deki kısımda ise istream ve ostream operatörleri test ediliyor.

Daha önce açtığımız objeyle (us) >> operatörünü çağırarak sırayla kullanıcı bilgilerini girdikten sonra << operatörü sayesinde bu girdiğimiz bilgiler objeyle çağırılıyor.

```
--User Info--
ID: 123
Name: Ahmet
Surname: Yilmaz
Age: 20
Week: 1
Weight: 70
-Constructor has been tested-

--User Info--
ID: 123
Name: selim
Surname: aygun
Age: 21
Week: 1
Weight: 70
-Copy Constructor has been tested-
```

Sekil-13

Sekil-13'te, yeni oluşturulan objeye (us2) hazır bilgiler girilerek constructor oluşturuluyor ve ostream operatörüyle ekrana basılıyor. Ardından user sınıfından yeni bir obje (us3) oluşturularak Sekil-13'te oluşturulan obje (us), bu objeye (us3) kopyalanıyor.

```
--User Info--
ID: 123
Name: selim
Surname: aygun
Age: 21
Week: 1
Weight: 70
-Assignment Constructor has been tested-

--User Info--
ID: 246
Name: selimselim
Surname: aygunaygun
Age: 42
Week: 2
Weight: 140
+Operator has been tested
```

Sekil-14

Sekil-14'te, Sekil-13'te oluşturulan objeler kullanılarak atama işlemi yapılıyor. Sekil-13'te ikinci oluşturduğumuz obje(us3), ilk oluşturduğumuz objeye(us2) atanıyor. Sekil-3'te ikinci oluşturduğumuz obje(us2) Sekil-11'de oluşturulan objeden(us) kopyalandığı için sonuç bu şekilde çıkıyor.

Alt kısımdaki toplama operatörü kısmında ise Sekil-13'te oluşturulan objeler(us2,us3) toplanıp Sekil-11'de oluşturulan objeye(us) eşitlenip ekrana bastırılıyor.

```
*****<Meals Class Tests>*****
size: 0
kcal: 0
-Default constructor has been tested-

size: 2
kcal: 400
-Constructor has been tested-
size: 2
kcal: 400
+Operator has been tested
```

Sekil-15

Sekil-15'teki kısımda yemek sınıfları olan Breakfast, Lunch ve Dinner sınıfları test edilmektedir. İlk önce Breakfast sınıfından obje oluşturularak default constructor çağırılmıştır. Ardından 2 yani medium boy öğün gönderilerek bir constructor oluşturulmuş ve verileri default olan obje ile toplanıp ekrana bastırılmıştır.

```
Set size operator is testing
kcal: 600
number: 1
size: 3
kcal: 800
-Assignment Constructor has been tested-
```

Sekil-16

Sekil-16'daki kısımda Breakfast sınıfından Sekil-15'te oluşturduğumuz objeyle set_size() fonksiyonu parametresinde large bilgisi taşıyan 3 değeri gönderilerek kcal bilgisi 600'e eşitlenmiştir. Ardından Breakfast sınıfındaki aynı objeyle get_kcal() ile kcal bilgisi, get_number() ile bu sınıftan kaç öğün yenildiği bilgisi çağrılmıştır. Ardından yeni bir obje oluşturularak daha önce oluşturduğumuz objeye atanmıştır. Böylece assignment operatör test edilmiştir.

```
kcal: 800
Get function has been tested
size :1
kcal: 400
size :2
kcal: 600
-----
size :3
kcal: 1000
+Operator has been tested
```

Sekil-17

Sekil-17'deki kısımda Lunch sınıfından objeyle get_kcal() fonksiyonuyla kcal çağrılıp ekrana bastırılıyor.

Ardından Dinner sınıfından 2 obje oluşturulup şekilde görüldüğü üzere toplama operatörüyle toplanıyor ve ekrana bastırılıyor.

```
*****<Sports Class Tests>*****
minute: 30
kcal: 150
-Constructor has been tested-

minute: 0
kcal: 0
minute: 30
kcal: 150
-Assignment Constructor has been tested-

-Default constructor has been tested-

minute: 10
kcal: 70
+Operator has been tested
```

Sekil-18

Sekil-18'de Sport sınıflarındaki Tennis sınıfından objeyle constructor oluşturup operatörler şekildeki üzere test edilmiştir.

REFERANSLAR

Write the sources you used for this project as separately numbered.

Projeyi yaparken faydalandığınız kaynakları (kitap, web siteleri, videolar vb.) bu bölümde maddeler halinde birinci maddede gösterildiği gibi yazınız.

1. ... (erişim tarihi: 18.05.2021)
2. ... <https://www.softwaretestinghelp.com/cpp-errors/>
3. ... <https://stackoverflow.com/questions/16810485/cannot-convert-stdstring-to-const-char/16810526>
4. ... <https://drawio.freshdesk.com/support/solutions/articles/16000042487-why-text-in-exported-svg-images-may-not-display-correctly>