

# Latihan 1

- a. Tuliskan realisasi dari fungsi **countFactorOfX** berikut ini:

```
countFactorOfX :: Int -> [Int] -> Int
{- countFactorOfX n l mengembalikan banyaknya kemunculan bilangan yang
   merupakan faktor dari n pada l -}
```

- b. Tuliskan realisasi dari fungsi **delNthElmt** berikut ini:

```
delNthElmt :: Int -> [Char] -> [Char]
{- delNthElmt n l menghilangkan elemen ke-n dari l.
   Asumsi: n lebih kecil atau sama dengan jumlah elemen l;
   l tidak kosong -}
```

# Latihan 2

Tuliskan definisi, spesifikasi, dan realisasi dari:

- a. Fungsi **sumIsiList** menghitung hasil penjumlahan dari seluruh elemen sebuah list of integer *l* yang tidak kosong.
- b. Fungsi **filterGanjil** melakukan filtering terhadap sebuah list of integer *li* sehingga menghasilkan list dengan elemen yang hanya terdiri atas bilangan ganjil yang muncul di *li*. Diasumsikan semua elemen *li* adalah bilangan integer positif atau 0. *li* mungkin kosong.

## Latihan 3

Tuliskan definisi, spesifikasi, dan realisasi dari fungsi **isEqFront** menerima masukan 2 buah list of character yang tidak kosong, misalnya T1 dan T2 dan menghasilkan true jika potongan awal list T2 mengandung T1 (dengan panjang dan urutan karakter yang sama). Banyaknya elemen T2 selalu lebih dari atau sama dengan T1.

Contoh:

T1: ['a', 'b', 'c']	T2: ['a', 'b', 'c', 'd', 'e']	Hasil: true
T1: ['a', 'b', 'c']	T2: ['a', 'b', 'c']	Hasil: true
T1: ['a', 'b', 'c']	T2: ['a', 'b', 'a', 'b', 'c', 'd']	Hasil: false
T1: ['a', 'b', 'c']	T2: ['a', 'b', 'd', 'a', 'b', 'c']	Hasil: false

# Latihan 4

Tuliskan realisasi dari fungsi **isOrdered** berikut ini:

**isOrdered :: [Int] -> Bool**

{- isOrdered l menghasilkan true jika elemen-elemen pada l terurut membesar, false jika tidak.  
Prekondisi: l tidak kosong -}

# Latihan 5

Tuliskan realisasi dari fungsi **mergeList** berikut ini:

```
mergeList:: [Int] -> [Int] -> [Int]
{- mergeList li1 li2 menghasilkan list of integer yang
   merupakan hasil penggabungan li1 dan li2, dan tetap
   terurut membesar.
   Prekondisi: li1 dan li2 adalah list terurut membesar
   dan mungkin kosong -}
```

# Latihan 6

Tuliskan realisasi dari fungsi **splitList** berikut ini:

```
splitList:: [Int] -> ([Int],[Int])  
{- splitList li menghasilkan 2 list of integer, list  
  pertama memuat bilangan positif dan 0 yang merupakan  
  elemen dari li (dengan urutan kemunculan yang tidak  
  berubah), sedangkan list kedua memuat bilangan negatif  
  elemen li.  
  Prekondisi: li mungkin kosong -}
```