

## Megjegyzések (if, mértékegységek)

### „Szalámszeletelés”

Tekintsük azt a feladatot, amiben azt kell megvizsgálni, hogy egy adott életkorban mit vezethet az illető: semmit, segédmotorkerékpárt vagy autót is.

Egy lehetséges megoldás:

```
if (kor < 14) {  
    System.out.println("semmit nem vezethet");  
} else if (kor >= 14 && kor < 17) {  
    System.out.println("segédmotorkerékpárt vezethet");  
} else if (kor >= 17) {  
    System.out.println("autót is vezethet");  
}
```

Itt már tudjuk,  
hogyan kor >= 14,

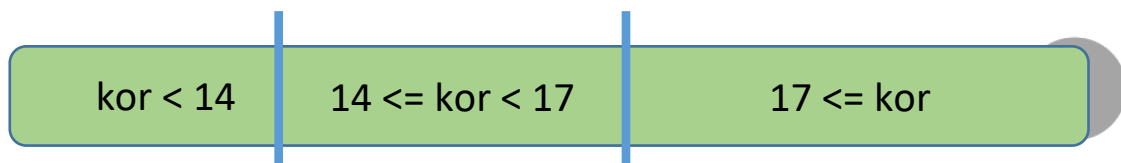
Ezért azt a feltételrészét, hogy `kor >= 14` már nem is kellene beírni.

Ugyanez a helyzet a harmadik feltétellel. a `kor >= 17` fölösleges. Ekkor azonban már semmi nem marad. Elég ekkor az `else`-ág. (Feltételezzük, hogy csak helyes adatokat adhatunk meg.)

Nézzük, hogy alakíthatjuk át:

```
if (kor < 14) {  
    System.out.println("semmit nem vezethet");  
} else if (kor < 17) {  
    System.out.println("segédmotorkerékpárt vezethet");  
} else {  
    System.out.println("autót is vezethet");  
}
```

A tartományok ilyenfajta kezelése nagyon hasonlít a szalámszeleteléshez:



Először levágjuk a „`kor < 14`” szeletet. Ami marad, arra biztosan igaz, hogy `kor >= 14`. Ebből

utána a „`kor < 17`” feltétellel levágunk még egy darabot (a matematikai  $14 \leq kor < 17$ -et), ami marad, az már biztosan `kor >= 17` (mindenféle egyéb feltétel nélkül).

## Hogyan ábrázoljunk mértékegységeket?

Leginkább sehogy.

Ha beolvasunk egy számot, az csak a mi (és a felhasználónk) fejében kapcsolódik össze a mértékegységével.

Ettől függetlenül érdemes mindig tisztában lenni, az adott szám milyen mértékegységben értelmezett. Történt már űrkutatási katasztrófa amiatt, hogy a vezérlőprogram egyik fele az adott számot SI-ben értette, a másik része pedig angolszász mértékegységűnek tekintette. ([Mars Climate Orbiter](#))

## Megjegyzés az if-fel kapcsolatban

```
if (a == 2) {  
    c = true;  
} else {  
    c = false;  
}
```

helyett használhatjuk röviden:

```
c = a==2;
```

vagy, mivel a fentit nem mindenki érti ránézésre:

```
c = (a == 2);
```

A kettő ugyanazt jelenti (lásd precedencia-tábla), de a lenti jobban olvasható.

## Másik megjegyzés az if-fel kapcsolatban

```
boolean b = ...;  
if (b == true) {  
    // valami  
}
```

helyett írjuk inkább ezt:

```
boolean b = ...;  
if (b) {  
    // valami  
}
```

A `b` értéke ugyanis logikai (igaz vagy hamis), az `if` pont egy logikai értéket vár, azaz neki a `b` önmagában is elég lesz.

Illetve ugyanez `false`-szal:

```
boolean b = ...;
if (b == false) {
    // valami
}
```

helyett írjuk inkább ezt:

```
boolean b = ...;
if (!b) {
    // valami
}
```

### Bonyolult feltételek megalkotása

Ha sok, akár egymásra ható részfeltételből akarunk egy feltételt összerakni, akkor a józan paraszti éssen lapuló metódus mellett van még két fogódzkodónk. Az egyik az **optimista**, a másik a **pesszimista** hozzáállást testesíti meg. Az optimista hozzáállás esetén megfogalmazzuk azokat az eseteket, melyek kedvezőek, és a többi lesz a „hamis”, míg a pesszimista esetben azt fogalmazzuk meg, mikor nem lesz jó a feltétel, és a többi esetben lesz igaz. (Az optimista és pesszimista, mint megfogalmazás természetesen nem értékrendi indíttatású, adott esetben a pesszimista hozzáállás egyszerűbb programot eredményez, akkor használjuk azt!)

#### Optimista megfogalmazás

Egy társasházban a felújítási munkákat hétköznap 9-17, szombaton 10-13 óráig engedélyeznek. Ha tudjuk a napot (1-7) és az órát (0-23), akkor kérdés, hogy az adott időszakban lehet-e felújítást végezni.

```
int day = ...;
int hour = ...;
if (day >= 1 && day <= 5 && hour >= 9 && hour < 17 || day == 6 &&
hour >= 10 && hour < 13) {
    System.out.println("Szabad");
} else {
    System.out.println("Nem szabad!");
}
```

### Pesszimista megfogalmazás

Egy találkozóra szeretnénk időpontot megbeszélni. Nem jó nekünk a hétfő, kedden 16 órától valamint csütörtökön 10-ig.

```
int day = ...;
int hour = ...;
if (!(day == 1 || day == 2 && hour >= 16 || day == 4 && hour < 10))
{
    System.out.println("Jó lesz az időpont!");
} else {
    System.out.println("Sajnos nem lesz jó!");
}
```

Azaz megfogalmaztuk a számunkra rossz eseteket, majd vettük az egész ellentétét – (!) operátor –, hogy megkapjuk a jó időszakokat.