



## Гранање

Када желиш да опишеш поступак за решавање неког проблема или обављање некаквог посла често се дешава да постоје одређени кораци које је могуће или потребно извести само ако је неки услов испуњен. На пример, припрема за одлазак на излет поред језера би могла да буде описана на следећи начин:

1. Припреми сендвиче
2. Набави сокове
3. Спакуј сокове и сендвиче у корпу
4. Понеси простирку
5. Ако су најављене падавине
  - a. Понеси кишобран
  - b. Понеси кабаницуако нису
  - c. Понеси купаћи костим
  - d. Понеси патике за воду
6. Понеси лопту за одбојку

Као што видиш због корака 5. постоје два начина на који се према датом поступку за припрему можеш спремити. На који начин ћеш то извести зависи од одговора на питање да ли су најављене падавине или не. Када поступак, тј. алгоритам стижања до одређеног циља, подразумева различите кораке у зависности од испуњености одређених услова тада кажемо да у алгоритму долази до **гранања**.

Покушај да даш свој пример ситуације из реалног живота у којој поступак стижања до циља или решења задатка зависи и од испуњености некаквих услова.

Како у реалном животу, тако и у програмирању, постоје ситуације у којима је у поступку решавања проблема за извршавање одређених корака потребно испитати да ли је одређен услов испуњен. За те потребе је уведена и посебна наредба. Тема ове лекције је управо гранање у Python програмском језику.

За почетак одгледај следећу видео-лекцију



[Python – наредбе гранања](#)



Да би се у језику Python описало условно извршавање неких наредби користи се **наредба if** која има следећи облик:

```
1 if uslov:  
2     naredba_1  
3     ...  
4     naredba_k
```

Примети да се након услова обавезно мора навести двотачка (карактер :) и да се наредбе које се извршавају условно морају мало увући (обично се то уради тако што се испред сваке наредбе откуцају три размака).

Најчешћи и најједноставнији облик услова јесу поређења неких величина. Рачунар уме да пореди величине (бројеве, али и ниске). За то се користе оператори, такозвани релацијски оператори, слични онима које си већ видео у математици.

#### ОПЕРАТОРИ ПОРЕЂЕЊА

- a < b**    проверава да ли је a мање од b
- a > b**    проверава да ли је a веће од b
- a >= b**    проверава да ли је a веће или једнако b
- a <= b**    проверава да ли је a мање или једнако b
- a == b**    проверава да ли је a једнако b

Резултат примене ових оператора је **тачно** или **нетачно** (кажемо да је резултат логичка тј. истинитосна вредност).

Хајде сада да решимо један једноставан проблем за који је потребно користити гранање.

#### Задатак 1.

Напиши програм који корисницима млађим од 15 година шаље посебну похвалу јер су кренули да програмирају веома рано.

Решење овог задатка би могло да изгледа овако:

```
1 godine = int(input("Koliko imate godina: "))  
2 if godine < 15:  
3     print("Tako ste mladi, a već programirate! Svaka čast!")
```



У преходном задатку команда за штампање се извршавала само у случају да је услов задовољен, док ако услов није задовољен сам програм неће урадити ништа додатно. Често се јавља потреба да је у случају да када неки услов није задовољен потребно извршити некакав други скуп корака. На пример, ако је корисник унео исправну лозинку треба му пожелети добродошлицу на сајт, а у супротном му треба јавити да унета лозинка није исправна. Такав облик организације извршавања програма се постиже **наредбом if-else** која у језику Python има следећи облик:

```
1 if uslov:  
2     naredba_1  
3     ...  
4     naredba_m  
5 else:  
6     naredba_1  
7     ...  
8     naredba_n
```

Примети да је двотачку потребно навести и иза else, као и да су оба блока наредби увучена.

### Задатак 2.

Како би претходни програм требало да изгледа, ако би онима који имају 15 или више година било потребно рећи да никада није касно да крену са учењем програмирања?

Решење би могло да изгледа овако:

```
1 godine = int(input("Koliko imate godina: "))  
2 if (godine < 15):  
3     print("Tako ste mladi, a vec programirate! Svaka cast!")  
4 else:  
5     print("Nikada nije kasno da se krene sa učenjem  
    programiranja!")
```

Дакле, постоји када постоји услов који може и не мора бити испуњен и када се у зависности од његове испуњености извршава један или други скуп наредби, на месту испитивања услова у програму се јавља такозвано **гранање**. Једну грану чине кораци који се извршавају уколико је услов испуњен, а другу кораци који се извршавају уколико услов није испуњен. Зато се сама наредба назива **наредбом гранања**.

**Задатак 3.**

Напиши програм који након учитавања једног броја исписује коментар да ли је унети број позитиван или не.

**Предлог решења**

```
1 broj = int(input("Unesi broj"))
2 if broj > 0 :
3     print("Broj je pozitivan")
4 else:
5     print("Broj nije pozitivan")
```

**Задатак 4.**

Напиши програм који испитује да ли је унети број паран или непаран.

Проверу да ли је један број дељив другим можемо извршити тако што израчунамо остатак при дељењу и упоредимо га са нулом. Подсетимо се, о целобројном дељењу је већ било речи на ранијим часовима и тада смо рекли да се остатак може израчунати коришћењем оператора %.

**Предлог решења**

```
1 broj = 17
2 if broj % 2 == 0:
3     print("Broj je paran")
4 else:
5     print("Broj je neparan")
6
```



Поређење се може вршити и над нискама. Погледај наредни задатак.

**Задатак 5.**

У бајци браће Грим, патуљак се звао Цвилидрета и девојка је требало да погоди његово име. Ако принцеза промаши, потребно је исписати јој поруку да, нажалост, није погодила.

Предлог решења

```
1 ime = input("Kako se zove patuljak: ")
2- if ime != "Cvilidreta":
3     print("Nažalost nisi pogodila :(")
```

**Задатак 6.**

Напиши програм који проверава да ли је корисник научио колико бајтова постоји у једном килобајту.

Покушај да решиш овај задатак. Задатак можеш пронаћи и проверити сопствено решење у окружењу [интерактивног уџбеника](#).



## Комбиновање више логичких услова (логички оператори)

Постоје ситуације у којима сам услов који мора бити испуњен да би одређени кораци били извршени није могуће описати једноставним поређењем две вредности, већ је потребно проверити испуњеност више једноставних услова. На пример, учитељица жели да поклони књигу свим оним ученицима који су имали примерно владање током године и који су били одлични ђаци или су учествовали у украшавању учионице. И у говорном, а и у програмским језицима се за конструкцију сложених услова користе везници којима се се везију једноставнији услови. У програмским језицима се они називају **логичким операторима**.

Једноставније услове комбинујемо речима **и**, **или** и **не**.

- Сложени услов облика

### **услов1 и услов2**

је испуњен само ако су оба услова, *услов1* и *услов2* испуњени.

На пример, да би реченица грмело је и севало била тачна, потребно је да је грмело и да је севало. Реч и се у језику Python записује помоћу речи **and**.

- Да би сложени услов облика

### **услов1 или услов2**

био испуњен довољно је да је бар један од услова *услов1* и *услов2* испуњен.

На пример, реченица “ићи ћу за викенд у позориште или у биоскоп” је тачна ако одеш само у позориште, само у биоскоп, али и ако одеш и у позориште и у биоскоп. Реч или се у језику Python записује помоћу речи **or**.

- Да би услов облика

### **не услов**

био испуњен потребно је да услов *услов* не буде испуњен.

На пример, реченица “Данас не пада киша” је тачна само ако реченица “Данас пада киша” није тачна. Реч не се у језику Python записује помоћу речи **not**.

### Задатак 1.

Напиши програм у који за унети број година исписује коментар да је особа са датим бројем година тинејџер или не, при чему се сматра да је особа тинејџер уколико има бар 12 и не више од 19 година.

### Предлог решења

```
1 broj = int(input("Unesi broj godina"))
2 if broj >= 12 and broj <= 19:
3     print("Ti jesi tinejdzer")
4 else:
5     print("Ti nisi tinejdzer")
```



Када се дефинишу сложени услови могуће је користити и заграде којима се, као и у аритметичким изразима, утиче на редослед извршавања операција. Тако на пример, нека је услов да учитељица награди ученика задат овако:

Ученик ће бити награђен ако

- је имао примерно владање током године и ако
- је одличан ђак или је учествовао у украшавању учионице.

Можеш уоштити да се услов састоји из три једноставна услова:

услов1 : има примерно владање

услов2 : одличан је ученик

услов3 : учествовао је у украшавању учионице

који би на следећи начин морали да буду повезани да би на исправан начин описали наведени сложени услов:

услов1 и (услов2 или услов3)

На који начин би се променио услов за награђивање ако би његов запис изгледао овако:

(услов1 и услов2) или услов3?

Који би ученик могао да се нађе на списку награђених који је формиран на основу овако описаног услова, а не би могао да се нађе на основу првобитно дефинисаног услова?

#### Одговор.

Да би ученик задовољио услов дефинисан са (услов1 и услов2) или услов3 довољно је да је учествовао у украшавању учионице. Ако такав ученик иначе нема примерно владање, према првобитно описаном услову он не би могао да буде награђен.

#### Задатак 2.

Кућни ред забрањује прављење буке пре 6 часова и између 13 и 17 часова, и након 22 часа. Напиши програм који радницима говори да ли у неком датом тренутку могу да изводе бучније радове.

Покушај да решиш овај задатак. Задатак можеш пронаћи и проверити сопствено решење у окружењу [интерактивног уџбеника](#).