

Numbri äraarvamismäng

Selle ülesande koostamisel võiks kasutada:

- muutujaid
- While tsükkel
- võrdlemist
- random moodulit
- tingimuslauseid (if, elif, else)
- print() funktsiooni
- input() funktsiooni

Ülesande mõte on koostada numברי äraarvamismäng. Iga mängu alguses peab Sinu kirjutatud programm valima suvalise numברי (nt 1-st 10-ni), mille programmi käivitaja peab ära arvama. Programm peab nii kaua küsima kasutaja käest uuesti numbrit, kuniks õige number on ära arvatud. Et mängu lihtsamini lahendatavaks teha, võiksid tuua sisse kaks võimalust: kui kasutaja poolt pakutud number on suurem kui arvuti poolt valitud number ja kui kasutaja poolt pakutud number on väiksem kui arvuti poolt valitud number. Nende kahe võimaluse realiseerimiseks võiks kasutada tingimuslauseid ja print() funktsiooni. Samuti võiks printida välja lause, mis annab kasutajale märku, et tema poolt pakutud number on õige.

Järgnevad näited võivad tulla kasuks ülesande lahendamisel:

Muutujate näide:

```
1
2      # Salvestan väärtused muutujatesse (int ja string - muutuja andmetüübid)
3
4      variable_one = 1
5      variable_two = "one"
```

Tingimuslausete näide:

```
2      # Kasutan kahte erinevat tingimuslauseid
3
4      variable_one = 1
5
6      if variable_one == 1: # kui variable_one on võrdne ühega, siis ...
7          print(1)
8      elif variable_one == "one": # kui eelmine if-lause polnud tõene ja variable_one on võrdne sõnega "one", siis ...
9          print("one")
10     else: # kui eelmised tingimuslauseid polnud tõesed, siis ...
11         print("Bye!")
```

Sellises olukorras prinditakse konsooli 1, kuna esimene tingimuslause osutub tõeseks ja ülejäänud tingimuslauseid ignoreeritakse.

While tsükli näide:

```

2     i = 0
3
4     while i != 10:
5         print(i)
6         i += 1 # muutuja i väärtust suurendatakse ühe võrra

```

While tsüklile antakse kaasa **boolean** väide, mida enne igat iteratsiooni kontrollitakse. Meie näites selliseks väiteks on **i != 0**. While tsükkel kestab, kuni kaasa antud väide on tõene ehk siin olukorras lõpeb tsükli tegevus, kui i on võrdne 10-ga.

Tsükli käigus prinditakse konsooli selline tulemus:

```

0
1
2
3
4
5
6
7
8
9

Process finished with exit code 0

```

Võrdlemise näide:

```

1
2     x = 0
3     y = 6
4
5     print(x < y)
6     print(x > y)
7     print(x == y)

```

Võrdluse puhul võrreldakse kahte väärtust ja antakse tulemuseks **True** või **False**. Praeguses näites prinditakse konsooli True, False, False.

Võrdlemiseks on meil operandid:

- < (väikesem kui)
- > (suurem kui)
- == (võrdne)
- != (mitte võrdne)

- >= (suurem või võrdne kui)
- <= (väiksem või võrdne kui)

input() näide:

```
2 # int() on input() ümber, kuna see näitab, et sisend peab olema integraal
3 question = int(input("How old are you? "))
4 # str() muudab muutuja question väärtuse sõneks
5 print("It must be nice to be " + str(question) + " years old!")
```

input() annab võimaluse programmi kasutajale sisendeid anda. Programmi käivitamisel prinditakse konsooli:

```
How old are you? 21
It must be nice to be 21 years old!

Process finished with exit code 0
```

Näitest on näha, et programmi kasutaja on andnud sisendiks arvu 21.

random näide:

Random moodulil on olemas selline funktsioon, mis valib suvaliselt ühe numbri sinu poolt määratud vahemikus.

```
import random

random_number = random.randint(1, 15)
print(random_number)
```

Siin näites valitakse suvaliselt üks number vahemikust 1-15. Igal uuel programmi käivitamisel valitakse uus number.