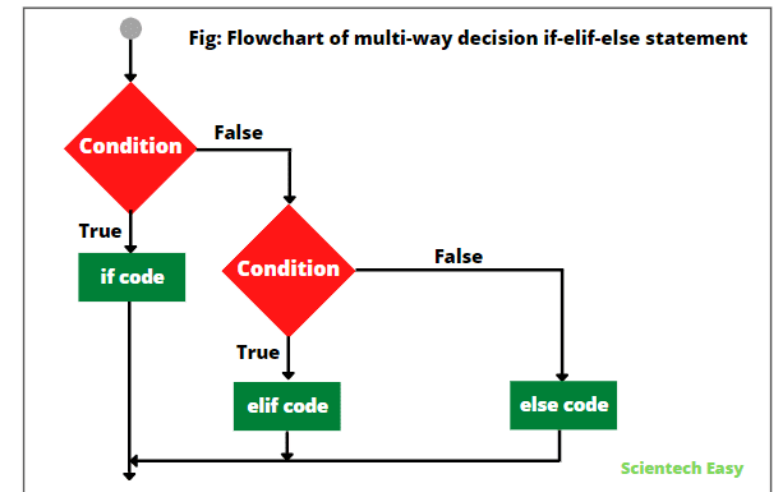


# **SAZAROJUMA KONSTRUKCIJAS**

**Markuss Sauka 2PT**

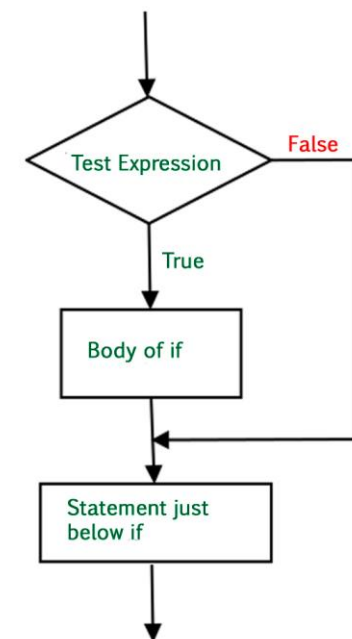
# KAS IR SAZAROJUMA KONSTRUKCIJAS?

Sazarojuma konstrukcija Python valodā ir nosacījumu izpildes struktūra, kas ļauj programmai pieņemt lēmumus un izpildīt dažādu kodu, balstoties uz kādu nosacījumu.



# KAD LIETO SAZAROJUMA KONSTRUKCIJA IF?

Sazarojuma konstrukciju if lieto, ja kāda programmas daļa jāizpilda tikai pie noteiktiem nosacījumiem. Ja tie neizpildās, tad if konstrukcijā iekļautās komandas tiek ignorētas un tiek izpildīta nākamā pī komanda, kas seko aiz if konstrukcijas.



# SAZAROJUMA KONSTRUKCIJA IF PIEMĒRS

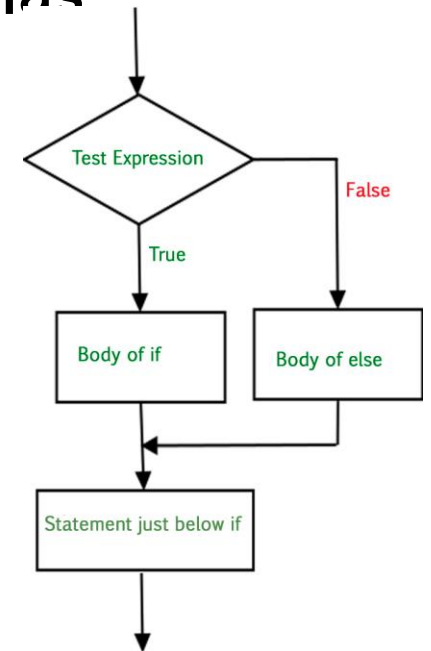
```
number = int(input('Enter a number: '))

# check if number is greater than 0
if number > 0:
    print(f'{number} is a positive number.')

print('A statement outside the if statement.')
```

# KAD LIETO SAZAROJUMA KONSTRUKCIJA IF ELSE?

Sazarojuma konstrukciju if...else lieto, ja kāda programmas daļa jāizpilda tikai noteiktu nosacījumu izpildīšanās gadījumā, bet neizpildīšanās gadījumā, jāizpilda cita programmas daļa.



# SAZAROJUMA KONSTRUKCIJA

## IF ELSE PIEMĒRS

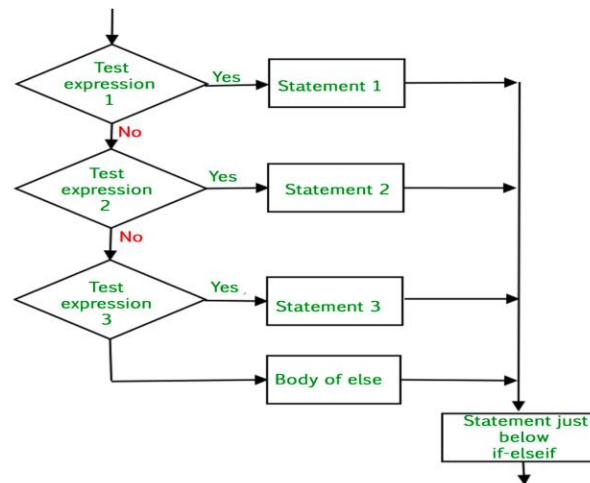
```
number = int(input('Enter a number: '))

if number > 0:
    print('Positive number')
else:
    print('Not a positive number')

print('This statement always executes')
```

# KAD LIETO SAZAROJUMA KONSTRUKCIJA IF ELIF?

If else tiek izmantots, lai izpildītu koda bloku, izvēloties starp divām alternatīvām. Tomēr, ja ir jāizvēlas starp vairāk nekā divām alternatīvām, tad izmantojam if...elif...else.



# SAZAROJUMA KONSTRUKCIJA

## IF ELIF PIEMĒRS

```
number = -5

if number > 0:
    print('Positive number')

elif number < 0:
    print('Negative number')

else:
    print('Zero')

print('This statement is always executed')
```



# PRIEKŠROCĪBAS

- Vienkāršība un saprotamība
- Plaša pielietojamība
- Nav nepieciešamas papildu bibliotēkas



# TRŪKUMI

- Grūti pārskatāms pie daudzām izvēlēm
- Nav visefektīvākais risinājums sarežģītai loģikai
- Grūtāk testēt un uzturēt



# 1.UZDEVUMS

Lietotājs ievada veselu skaitli. Programmai jāizvada, vai tas ir pozitīvs, negatīvs, vai nulle.

# 1.UZDEVUMS

```
# Prasa lietotājam ievadīt skaitli un pārveido to uz veselu skaitli (int)
skaitlis = int(input("Ievadi skaitli: "))

# Pārbauda, vai skaitlis ir lielāks par 0
if skaitlis > 0:
    print("Skaitlis ir pozitīvs.")
# Ja nav lielāks par 0, pārbauda, vai ir mazāks par 0
elif skaitlis < 0:
    print("Skaitlis ir negatīvs.")
# Ja nav ne pozitīvs, ne negatīvs, tad tas ir nulle
else:
    print("Skaitlis ir nulle.")
```

# 2.UZDEVUMS

Lietotājs ievada atzīmi no 1 līdz 10. Programmai jāizdrukā šāds novērtējums:

- 9–10: "Teicami"
- 7–8: "Labi"
- 4–6: "Vidēji"
- 1–3: "Vāji"
- Citi skaitļi: "Nederīga atzīme"

# 2.UZDEVUMS

```
# Lietotājs ievada atzīmi, kura tiek pārveidota uz veselu skaitli
atzime = int(input("Ievadi atzīmi (1-10): "))

# Pārbauda, vai atzīme ir 9 vai 10
if 9 <= atzime <= 10:
    print("Teicami")
# Ja nav, pārbauda, vai ir 7 vai 8
elif 7 <= atzime <= 8:
    print("Labi")
# Ja nav, pārbauda, vai ir no 4 līdz 6
elif 4 <= atzime <= 6:
    print("Vidēji")
# Ja nav, pārbauda, vai ir no 1 līdz 3
elif 1 <= atzime <= 3:
    print("Vāji")
# Ja nav neviena no iepriekšējiem - ievade nav no 1 līdz 10
else:
```

# 3.UZDEVUMS

Lietotājs ievada veselu skaitli. Programmai jāpasaka, vai tas ir pāra vai nepāra skaitlis.

# 3.UZDEVUMS

```
# Lietotājs ievada veselu skaitli
skaitlis = int(input("Ievadi veselu skaitli: "))

# Izmanto dalījuma atlikumu, lai pārbaudītu, vai skaitlis ir pāra
if skaitlis % 2 == 0:
    print("Skaitlis ir pāra.")
# Ja atlikums nav 0, tad tas ir nepāra skaitlis
else:
    print("Skaitlis ir nepāra.")
```



# IZMANTOTIE AVOTI

- Skolo.lv(Inta Znotiņa Programmēšana (Python))
- Programiz(<https://www.programiz.com/python-programming/if-elif-else>)
- GeeksforGeeks(<https://www.geeksforgeeks.org/python/python3-if-if-else-nested-if-if-elif-statements/>)
- ChatGPT