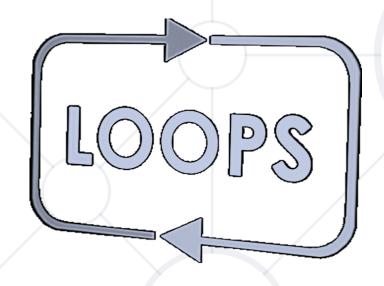
# Повторения (цикли)

Прости повторения с For-цикъл



СофтУни Преподавателски екип







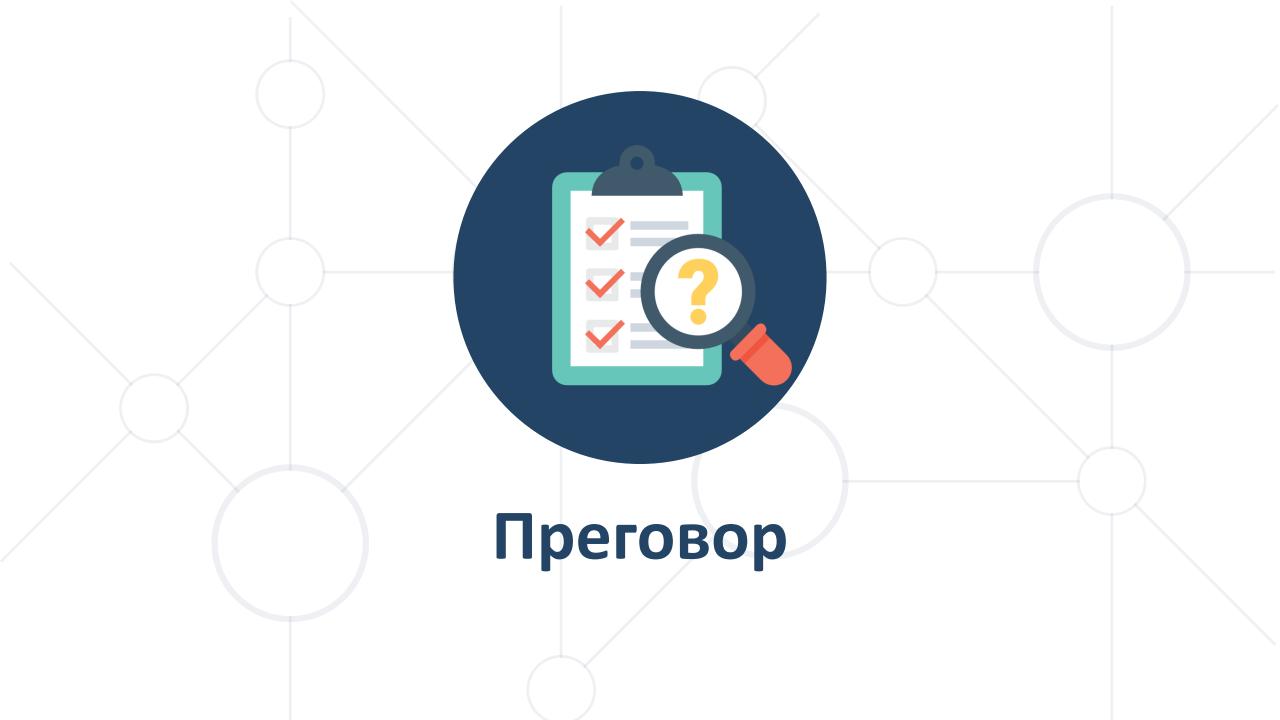
Софтуерен университет <a href="https://softuni.bg">https://softuni.bg</a>

# Съдържание



- 1. Преговор
- 2. Повторения на блокове код
- 3. Работа с по-сложни for-цикли
- 4. Работа с текст
- 5. Техники за използване на for-цикли



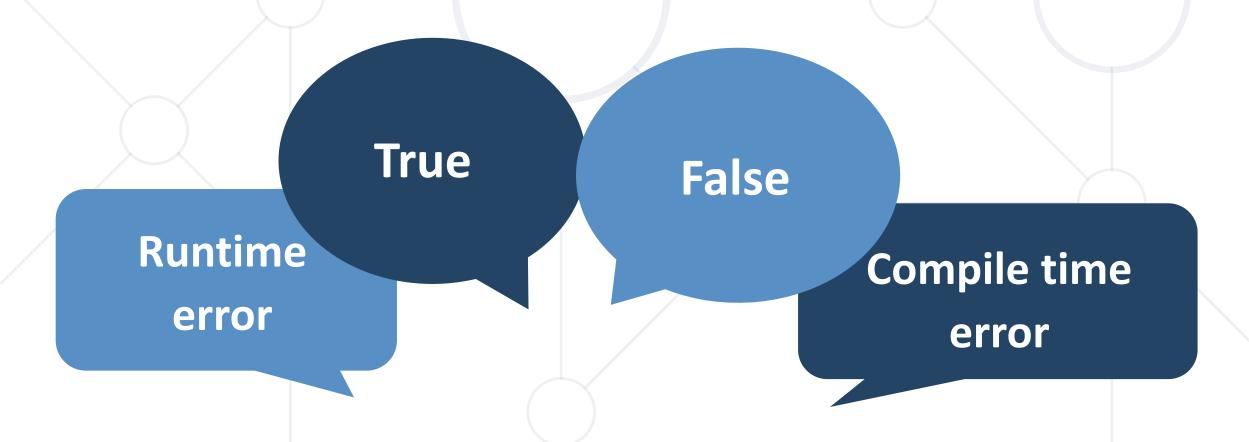




1. Какво ще се отпечата на конзолата, ако изпълним следната

команда

$$print(not (5 == 5) and (4 + 1 == 5))$$





2. Какво ще се отпечата на конзолата, ако изпълним следната

команда:

$$print(not (3 == 3) or (3 == 5))$$

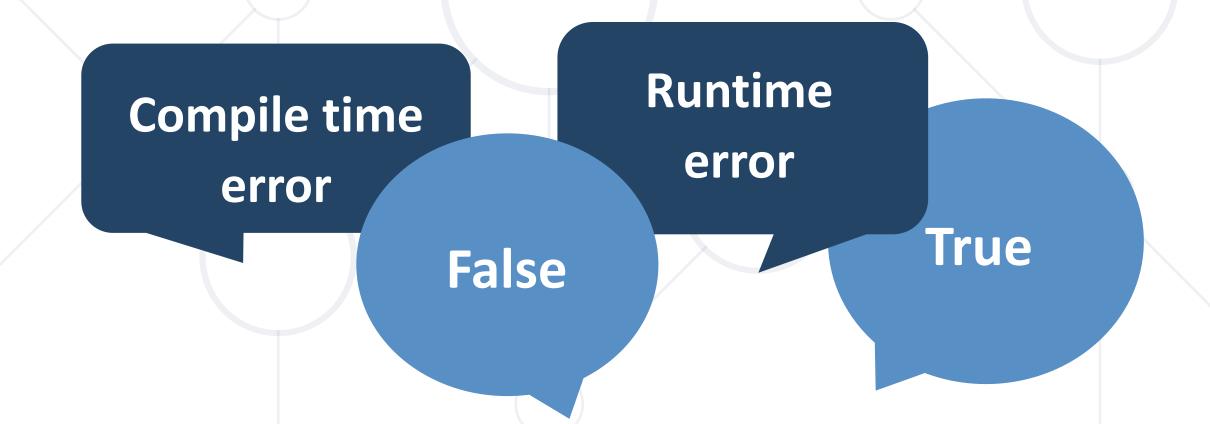




3. Какво ще се отпечата на конзолата, ако изпълним следната

проверка:

$$print(not (3 > 5) or (1 == 1))$$





4. Какво ще се отпечата на конзолата, ако изпълним следната

логическа проверка:

```
number = 101
if number >= 1:
    print("Larger than 1")
if number <= 101:
    print("Less than 101")
    print("Equal to 101")</pre>
```

Less than 101 Larger than 1 Equal to 101 Compile time error



5. Какво ще се отпечата на конзолата, ако изпълним следната логическа проверка:

```
role = "Administrator"
password = "SoftUni"
if role == "SoftUni":
    if password == "SoftUni":
        print("Welcome!")
```

**Runtime error** 

Welcome!

No output

Compile time error



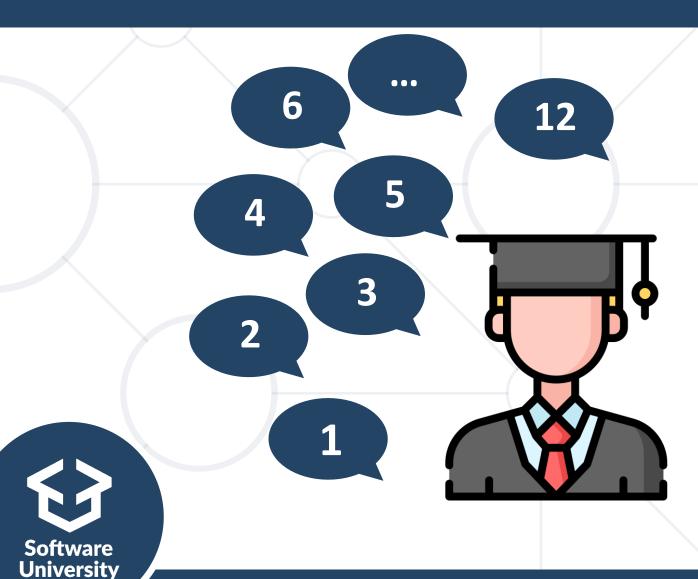
# Повторения на блокове код

Конструкция за For-цикъл

#### Какво е цикъл?



- Често ни се налага да повтаряме едно и също действие многократно
- Когато сме абитуренти броим до 12

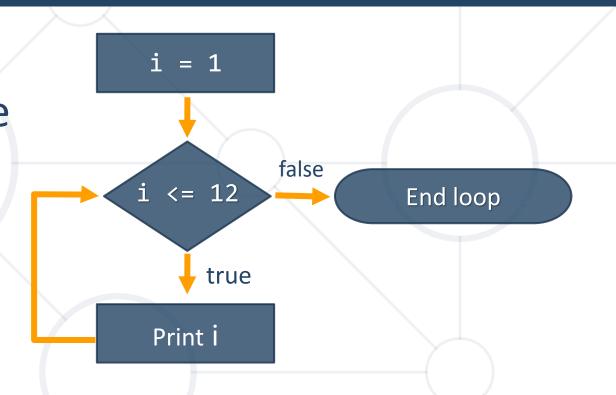


# Какво е цикъл? (2)



 Циклите в програмирането ни позволяват да повтаряме едни и същи действия определен брой пъти

for i in range(1, 13):
 print(i)



# For-цикъл – конструкция



 Можем да повтаряме действия до определен момент чрез for-цикли





# Работа с по-сложни For-цикли

Цикли със стъпка

#### For-цикъл със стъпка



 Можем да използваме стъпка ако искаме да променяме стойността на і със стойност различна от 1

```
for i in range(2, 11, 2):
print(i)
Задаване на стъпка 2
```

• Стъпката може да бъде негативна

```
for i in range(11, 0, -2):
print(i, end=" ") Задаване на стъпка -2
```

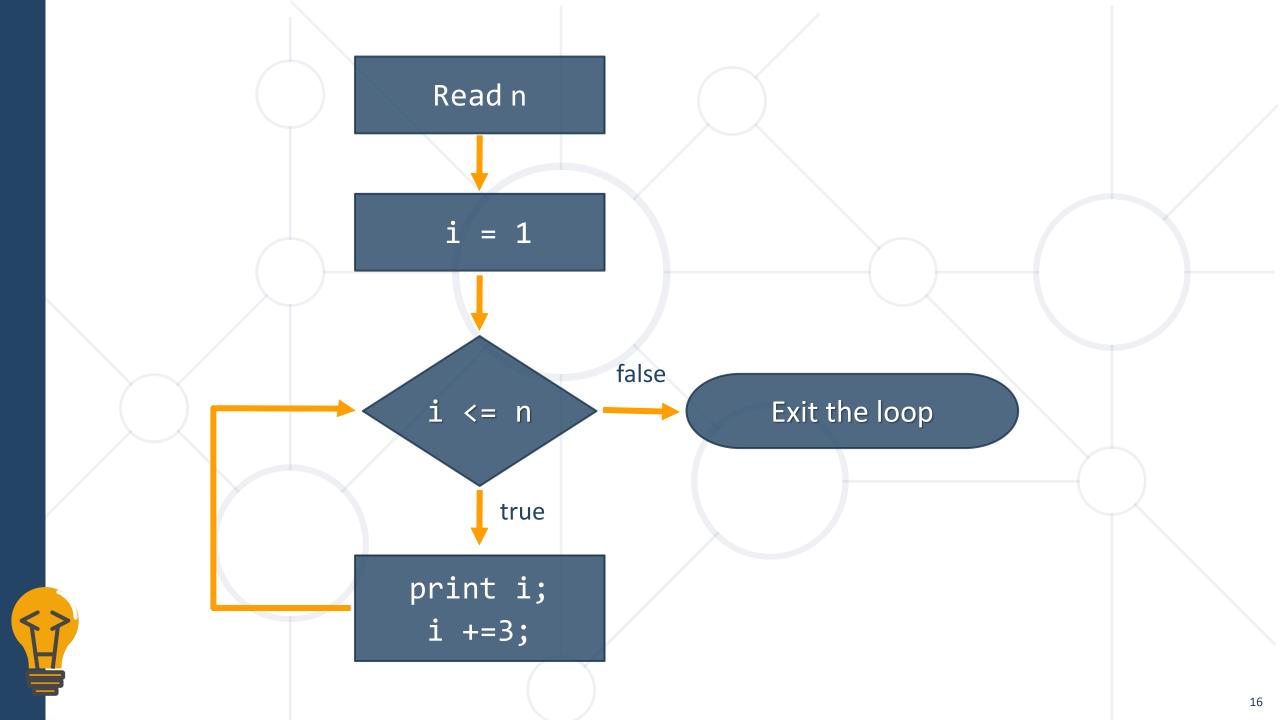
# Числата от 1 до N през 3 – условие



- Напишете програма, която:
  - Прочита цяло число n
  - Отпечатва числата от 1 до n със стъпка 3
- Примерен вход и изход:

10 | 1, 4, 7, 10





#### Числата от 1 до N през 3 – решение



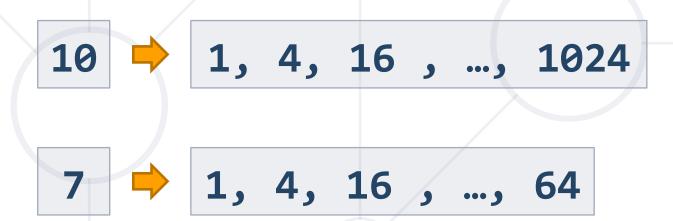
```
n = int(input())
for i in range(1, n + 1, 3):
print(i)

Задаване на стъпка 3
```

# Четни степени на 2 – условие



- Напишете програма, която:
  - Прочита цяло число n
  - Отпечатва четните степени на 2 до 2<sup>n</sup>: 2<sup>0</sup>, 2<sup>2</sup>, 2<sup>4</sup>, 2<sup>8</sup>, ..., 2<sup>n</sup>
- Примерен вход и изход:



#### Четни степени на 2 – решение



```
n = int(input())
num = 1
for i in range(0, n + 1, 2):
    print(num)
    num = num * 2 * 2

Ползваме
    стъпка 2
```

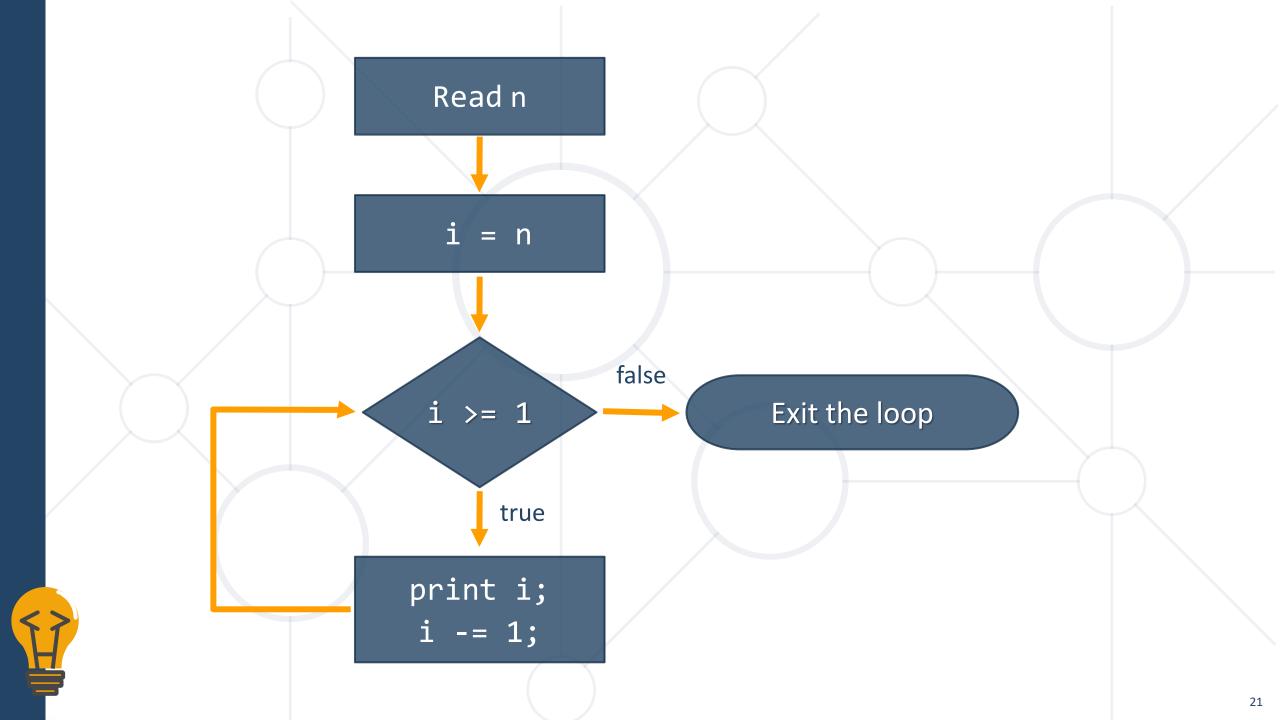
Тестване на решението: <a href="https://judge.softuni.bg/Contests/2417">https://judge.softuni.bg/Contests/2417</a>

# Числата от N до 1 в обратен ред – условие



- Напишете програма, която:
  - Прочита цяло число n
  - Отпечатва числата от n до 1 в обратен ред (стъпка -1)
- Примерен вход и изход:



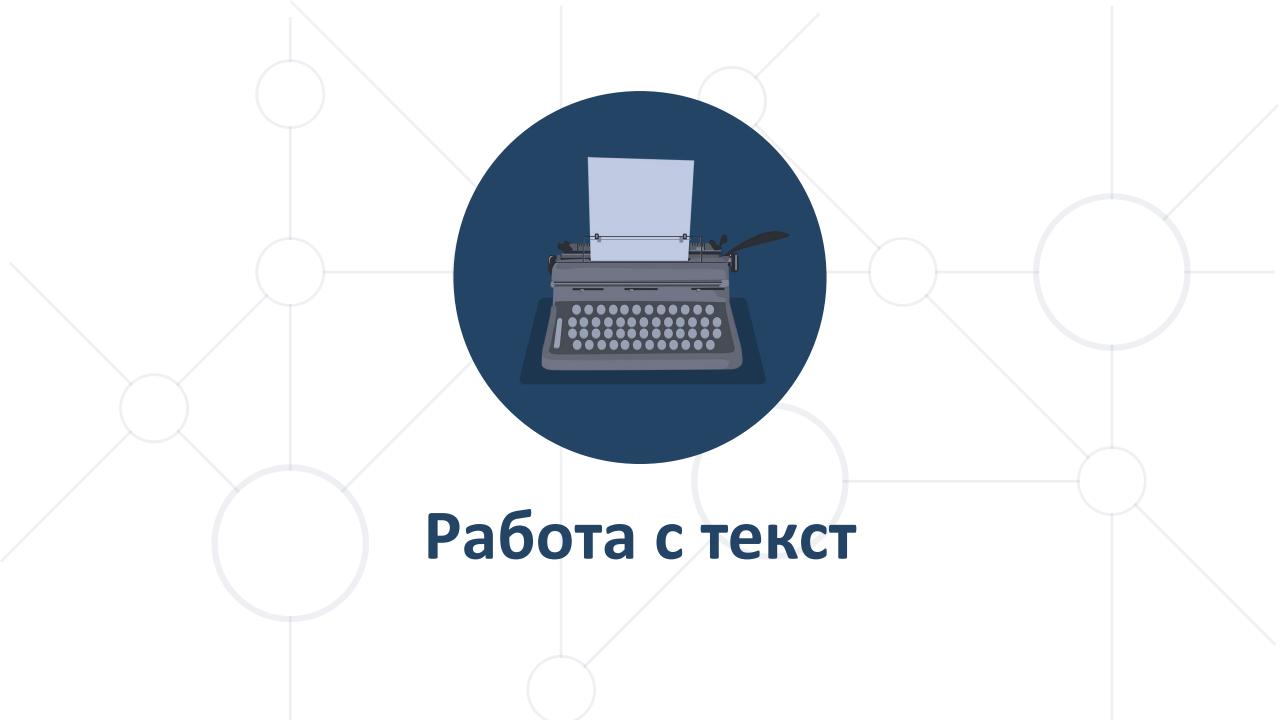


#### Числата от N до 1 в обратен ред – решение



```
n = int(input())
for i in range(n, 0,-1)
    print(i);
```

Обърнато условие: i >= 1



### Работа с текст



■ Можем да вземем дължината на текст

```
text = "SoftUni"
length = len(text) # 7
```

• Можем да вземем символ от текст по индекс

```
text = "SoftUni"
letter = text[4] # U
```

## Поток от символи – условие



- Напишете програма, която
  - чете текст(стринг)
  - печата всеки символ от текста на отделен ред



#### Поток от символи – решение



```
Взимаме дължината на текста
```

```
length = input()
for i in range(0,len(length)):
    print(length[i])
```

Взимаме всеки символ по индекс і

# Сумиране на гласни букви – условие



- Напишете програма, която:
  - Прочита от потребителя текст
  - Извежда сумата на гласните букви според таблицата по-долу:

а	е	-i	0	u
1	2	3	4	5

■ Примерен вход и изход:



6

$$(e+o = 2+4 = 6)$$

hi



3

$$(i = 3)$$

bamboo



9

$$(a+o+o=1+4+4=9)$$

beer



4

$$(e+e = 2+2 = 4)$$

#### Сумиране на гласни букви – решение



```
text = input()
sum = 0
for i in range(0,len(text)):
    if text[i] == "a":
        sum += 1
    if text[i] == "e":
        sum += 2
#TODO: make if statement for Letters 'i' 'o 'u'
print(sum)
```



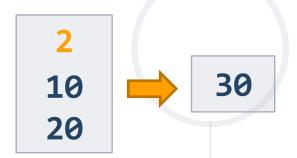
# Техники за използване на For-цикли

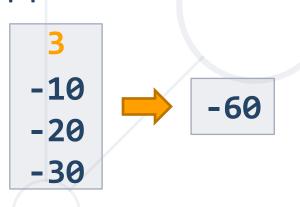
Задачи с цикли

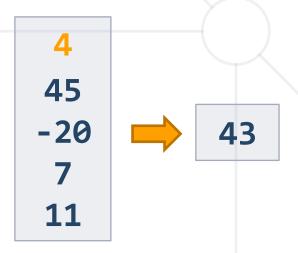
# Сумиране на числа – условие

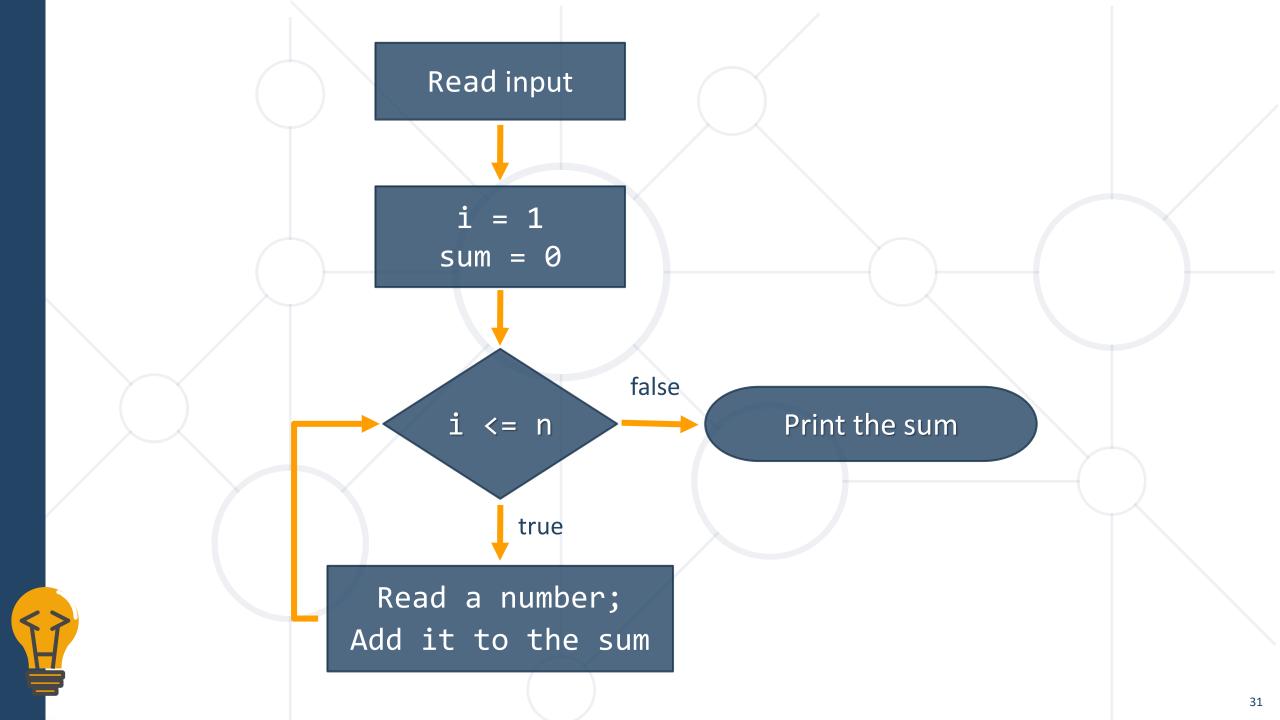


- Напишете програма, която:
  - Прочита цяло число n от потребителя
  - Прочита n последователни пъти числа и ги сумира
  - Извежда пресметнатата сума
- Примерен вход и изход:





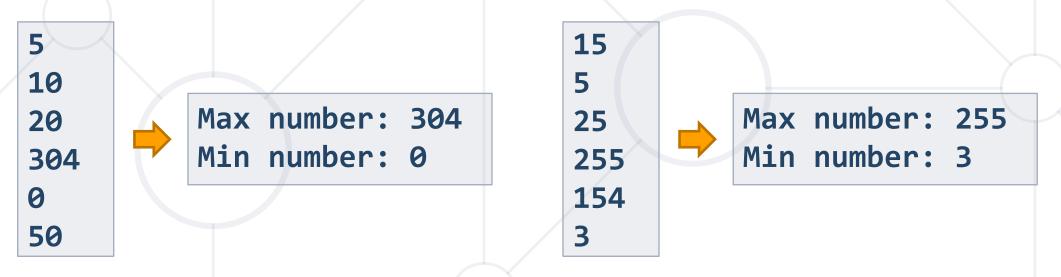


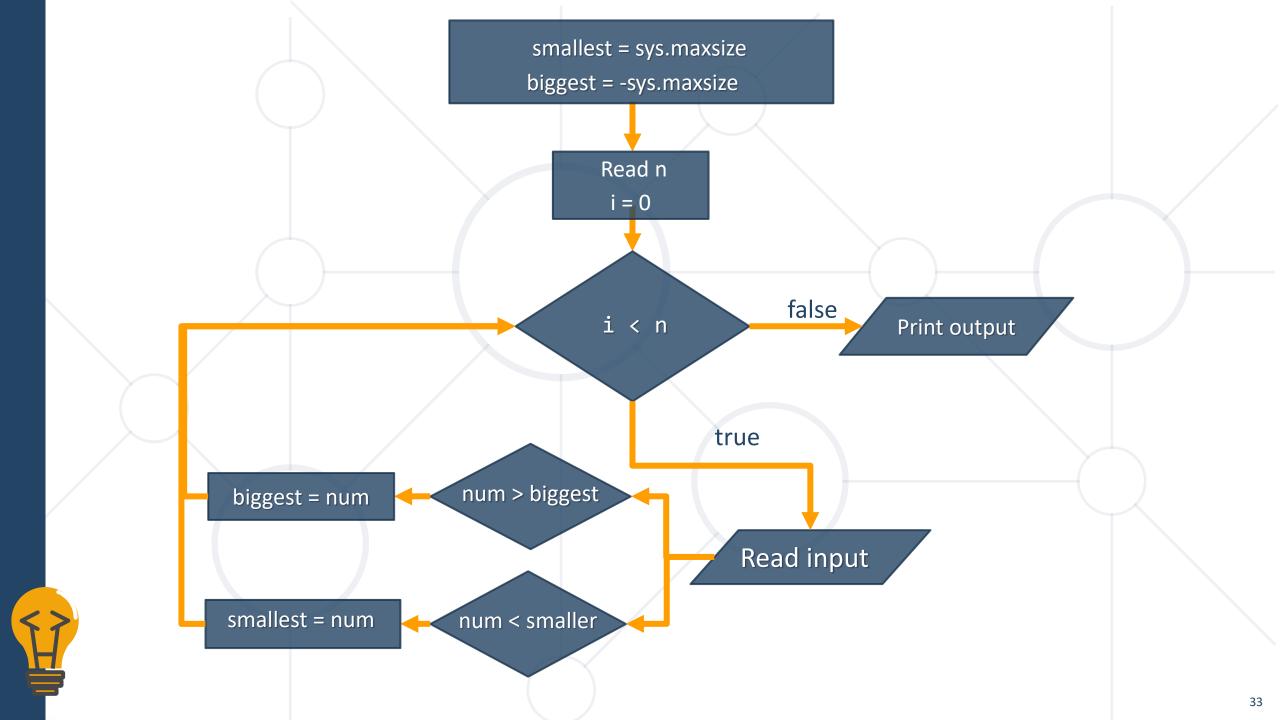


# Редица цели числа – условие



- Напишете програма, която:
  - Чете п на брой цели числа
  - Принтира най-голямото и най-малкото число
- Примерен вход и изход:





#### Редица цели числа – решение



```
import sys
smallest = sys.maxsize
biggest = -sys.maxsize
n = int(input())
for i in range(0, n):
    num = int(input())
    if num < smallest:</pre>
      smallest = num
    if num > biggest:
      biggest = num
print(f"Max number: {biggest}")
print(f"Min number: {smallest}")
```

13

# Лява и дясна сума – условие

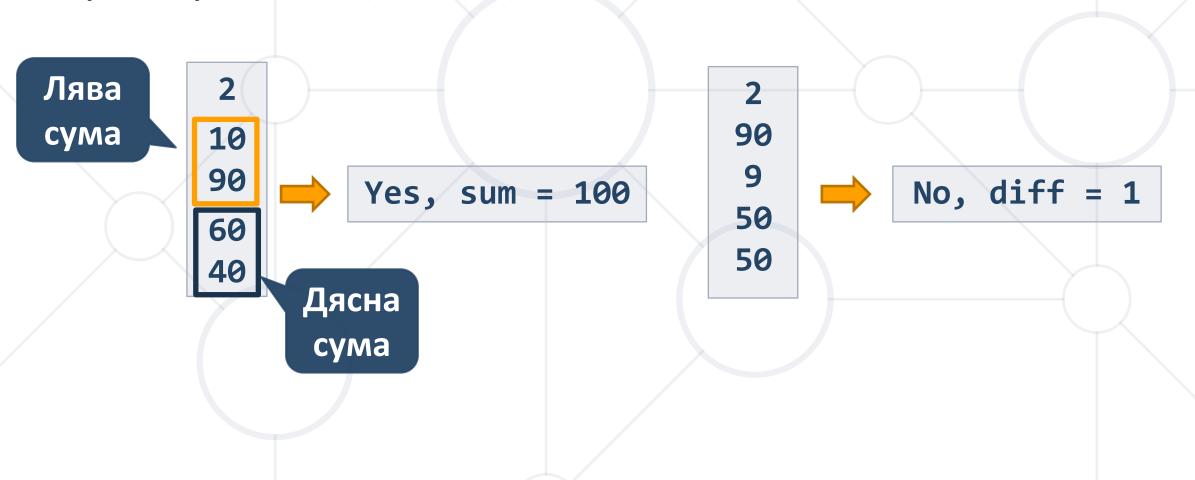


- Напишете програма, която:
  - Прочита цяло число n от потребителя
  - Прочита последователно 2\*n числа
  - Проверява дали сумите на левите n и десните n числа са равни
  - При равенство извежда "Yes" и сумата, в противен случай - "No" и разликата (изчислена като положително число)

# Лява и дясна сума – условие



■ Примерен вход и изход:



#### Решение: лява и дясна сума



```
int(input())
left sum = 0
for i in range(1, n + 1):
    left_sum = left_sum + int(input())
#TODO: read and calculate the right sum
if left sum == right sum:
    print(f"Yes, sum = {left sum}")
else:
    diff = abs(right_sum - left_sum)
    print(f"No, diff = {diff}")
```

Тестване на решението: <a href="https://judge.softuni.bg/Contests/2417">https://judge.softuni.bg/Contests/2417</a>

# Четна / нечетна сума – условие

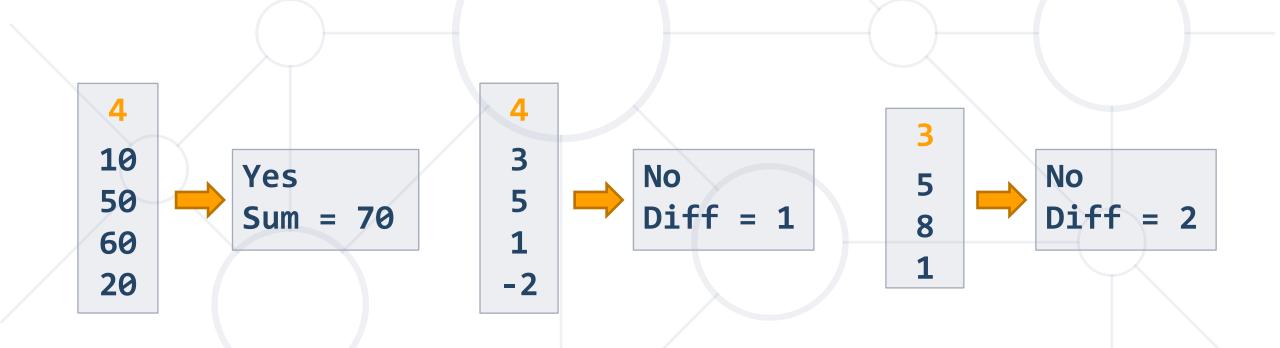


- Напишете програма, която:
  - Прочита цяло число(n) от потребителя
  - Прочита последователно n на брой числа
  - Проверява дали сумата на числата на четни позиции е равна на сумата на числата на нечетни позиции
  - При равенство печата "Yes" и сумата; иначе печата "No" и разликата (положително число).

# Четна / нечетна сума – условие



■ Примерен вход и изход:



# Решение: четна / нечетна сума



```
n = int(input())
odd sum = 0
even_sum = 0
for i in range(1, n + 1):
    element = int(input())
    if i % 2 == 0:
        even_sum += element
    else:
        odd sum += element
#TODO: print the sum / difference
```

# Какво научихме днес?



- Повторение на блок код с for-цикъл
- Цикли със стъпка
  - Цикли с увеличаваща стъпка
  - Цикли с намаляваща стъпка
- Вземане на символ по индекс от текст





# Въпроси?

















**Kids** 

## Лиценз



- Този курс (презентации, примери, демонстрационен код, упражнения, домашни, видео и други активи) представлява
   защитено авторско съдържание
- Нерегламентирано копиране, разпространение или използване е незаконно
- © СофтУни <a href="https://softuni.org">https://softuni.org</a>
- © Софтуерен университет <a href="https://softuni.bg">https://softuni.bg</a>



#### Обучения в Софтуерен университет (СофтУни)



- Софтуерен университет качествено образование, професия и работа за софтуерни инженери
  - softuni.bg
- Фондация "Софтуерен университет"
  - softuni.foundation
- Софтуерен университет @ Facebook
  - facebook.com/SoftwareUniversity
- Дискусионни форуми на СофтУни
  - forum.softuni.bg







