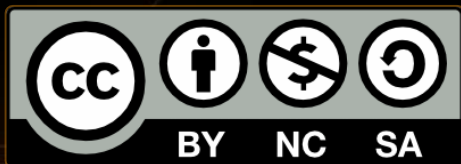


Повторения (цикли)

Прости повторения с for-цикъл



СофтУни

трейнърски екип

Софтуерен университет

<http://softuni.bg>



Основи на
програмирането



Имате въпроси?

sli.do

#pb-april

Съдържание

1. for-цикъл

- Конструкция

2. ASCII стойност и преобразуване на типове

3. Приложение





Повторения на блокове код

Конструкция за `for`-цикъл

Повторения (цикли) – for-цикъл

- В програмирането често пъти се налага да изпълним блок с команди **няколко** пъти
 - За целта използваме **цикли**

Начална стойност

```
for i in range(1, 10):
```

Крайна стойност

```
print('i = ' + str(i))
```

Ключова дума за конструкцията

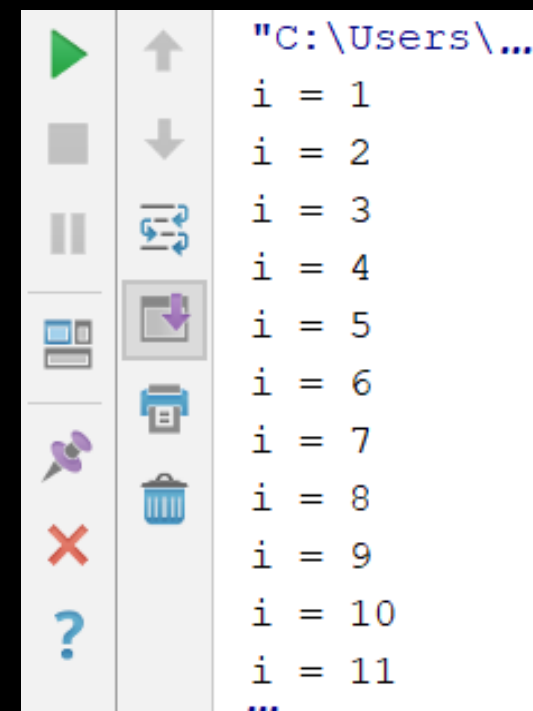
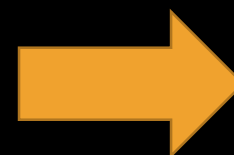
Тяло: блок от код за изпълнение (за повторение)

Числа от 1 до 100

- Напишете програма, която:
 - Извежда числата [1, 100], всяко на нов ред
- Решение:

Крайната стойност не е
включена в диапазона

```
for i in range(1, 101):  
    print('i = ' + str(i))
```



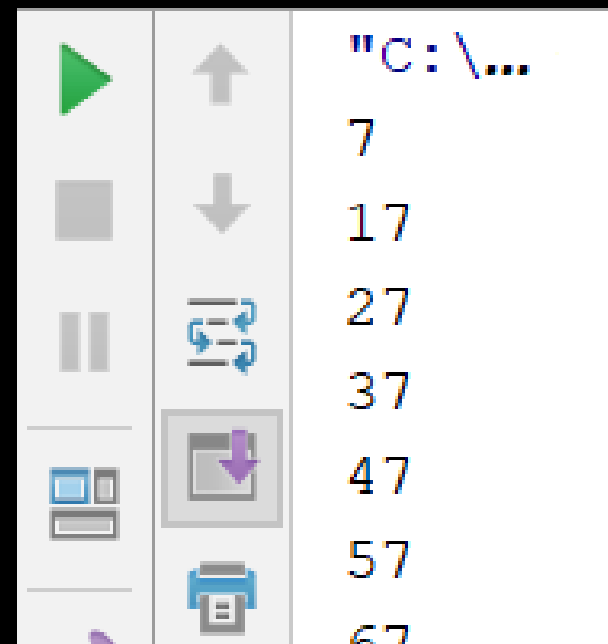
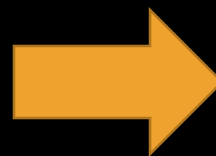
```
"C:\Users\...  
i = 1  
i = 2  
i = 3  
i = 4  
i = 5  
i = 6  
i = 7  
i = 8  
i = 9  
i = 10  
i = 11  
...
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#0>

Числа до 1000, завършващи на 7

- Напишете програма, която:
 - Извежда числата [1, 1000], които завършват на 7
- Решение:

```
for i in range(1, 1000):  
    if i % 10 == 7:  
        print(str(i))
```



Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#1>

ASCII таблица

- Символите, които използваме се представят като числа
 - Поместени са в **ASCII таблица**
- Примери (знак и неговата ASCII стойност):

a	→	97
---	---	----

A	→	65
---	---	----

@	→	64
---	---	----

+	→	43
---	---	----

Всички латински букви - условие

- Напишете програма, която:
 - Извежда буквите от латинската азбука: **[a, z]**
- Решение:

Функцията **ord** ни дава
числовата стойност на знака

```
for i in range(ord('a'), ord('z') + 1):  
    print(chr(i))
```

Функцията **chr**
ни дава знак от
числова стойност

Преобразуване на типове данни

- В програмирането можем да сменяме типа на данните чрез операцията "кастване"

- Примери

- От реално към цяло число:

```
a = (int)(5.66) # 5  
b = (int)(5.44) # 5
```

Получава се загуба на данни
(десетичната част от числото)

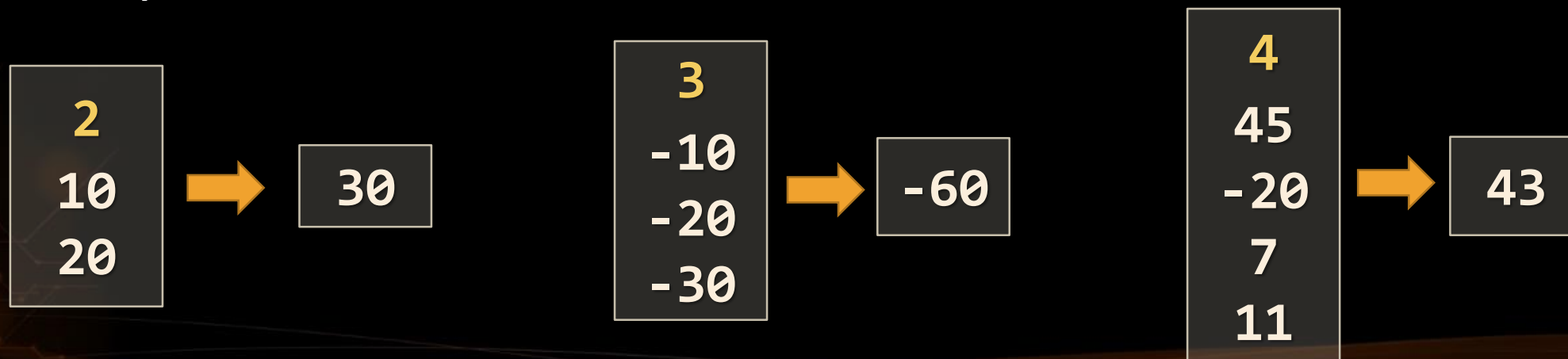
- От СИМВОЛ към ЧИСЛО и обратно:

```
a = (chr)(67)    # C  
b = ord('#')     # 35
```

Полученият резултат в `int` е
ASCII стойността на символа

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:

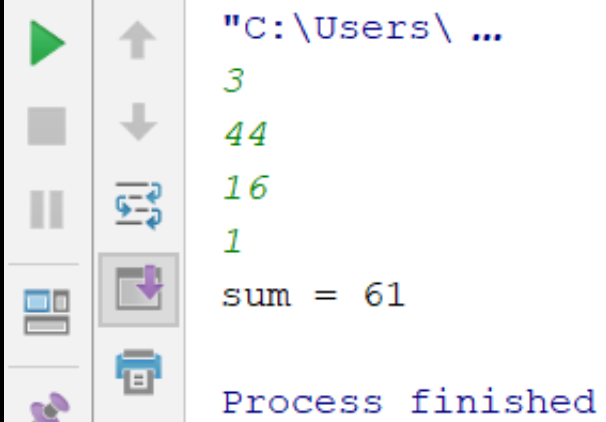
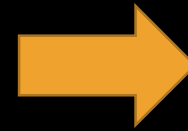


Сумиране на числа - решение

```
n = int(input())
sum = 0

for i in range (0, n):
    currentNum = int(input())
    sum = sum + currentNum

print("sum = " + str(sum))
```



```
"C:\Users\ ...
3
44
16
1
sum = 61
Process finished
```

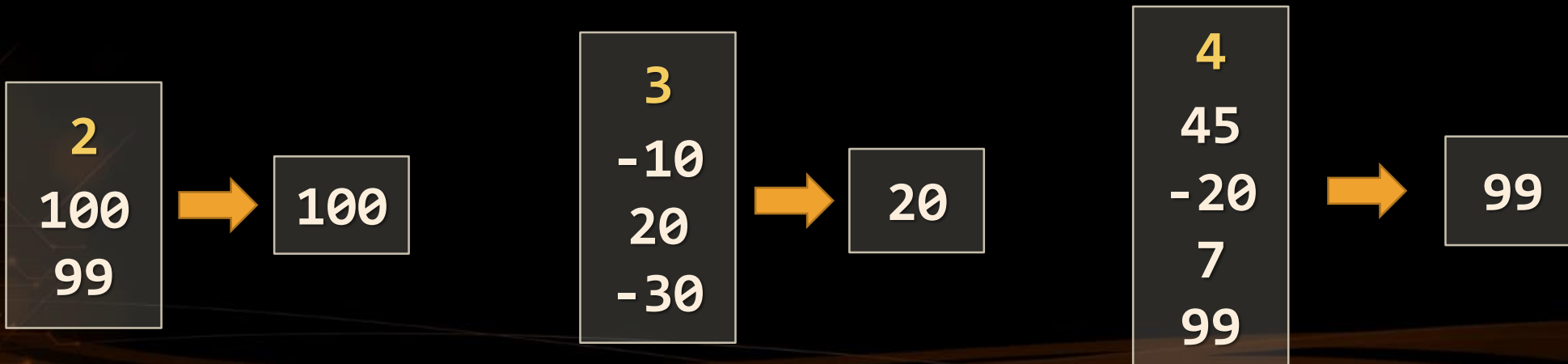
Можем да четем
данни в
конструкцията на
for-цикъл

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#3>

Най-голямо число - условие

- Напишете програма, която:
 - Прочита цяло число (**n**) от потребителя
 - Прочита (**n**) **последователни** пъти числа
 - Намира **най-голямото** измежду тях
- Примерен вход и изход:

5
3

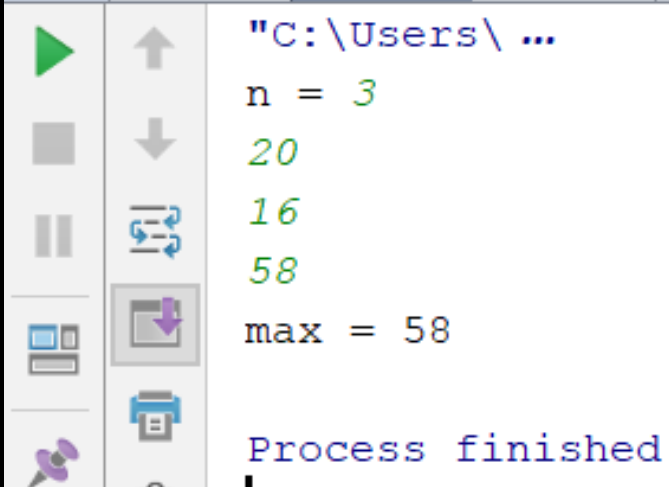
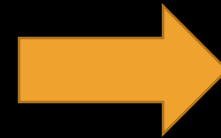


Най-голямо число - решение

```
n = int(input('n = '))
max = -1000000000000000

for i in range(n):
    num = int(input())
    if num > max:
        max = num

print('max = ' + str(max))
```

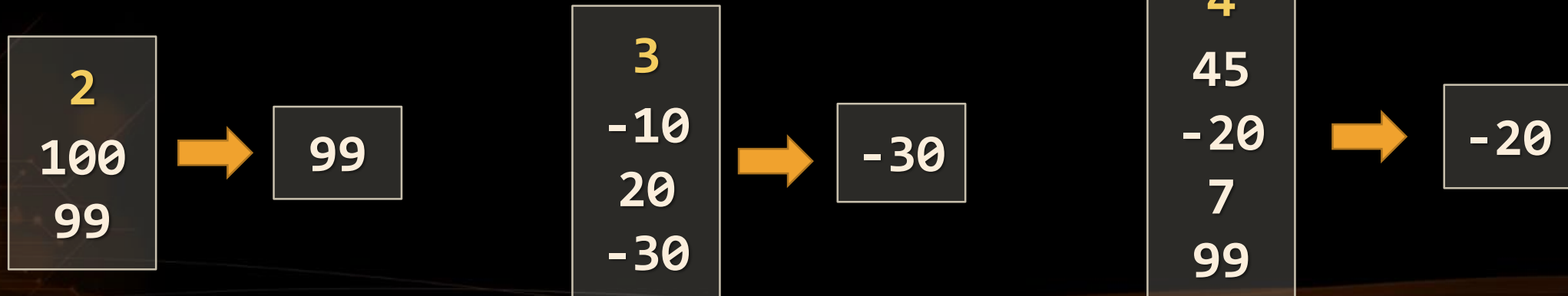


The screenshot shows a code execution interface. On the left is a toolbar with icons for running (green play button), stepping through (up/down arrows), pausing (two vertical bars), and other functions. The main area displays the output of the program: the path "C:\Users\ ..." is shown in blue; the input "n = 3" is in green; the subsequent inputs "20", "16", and "58" are in green; the output "max = 58" is in green; and the message "Process finished" is in blue at the bottom.

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#4>

Най-малко число - условие

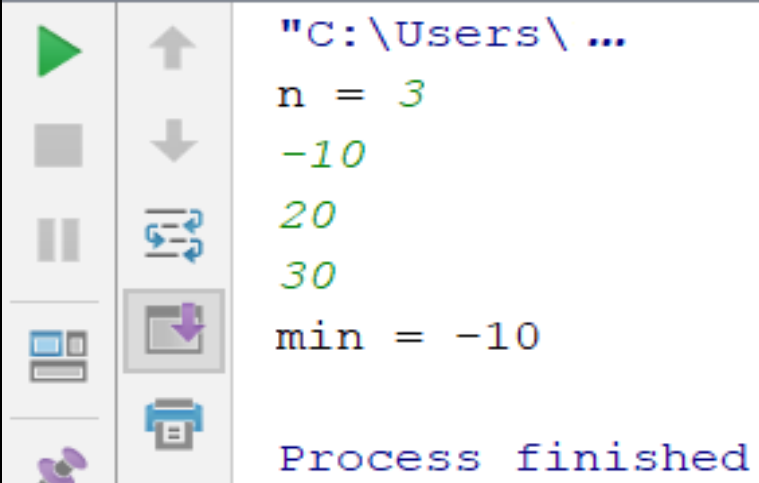
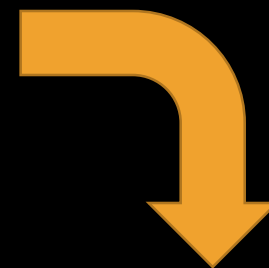
- Напишете програма, която:
 - Прочита цяло число (**n**) от потребителя
 - Прочита **n** последователни пъти числа
 - Намира **най-малкото** измежду тях
- Примерен вход и изход:



Най-малко число - решение

```
n = int(input('n = '))  
max = 1000000000000000
```

Use similar logic to previous problem



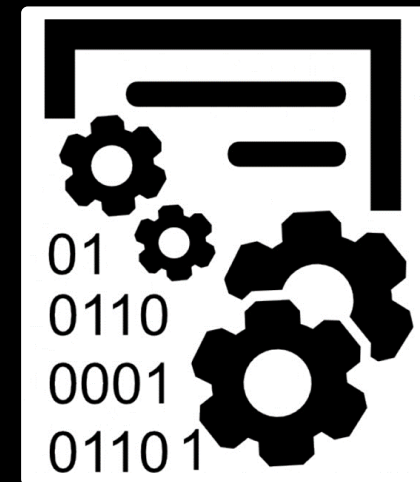
```
"C:\Users\ ...  
n = 3  
-10  
20  
30  
min = -10  
  
Process finished
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#4>



Повторения на блокове код

Работа на живо в клас (лаб)



Задачи с цикли

Техники за използване на for-цикли

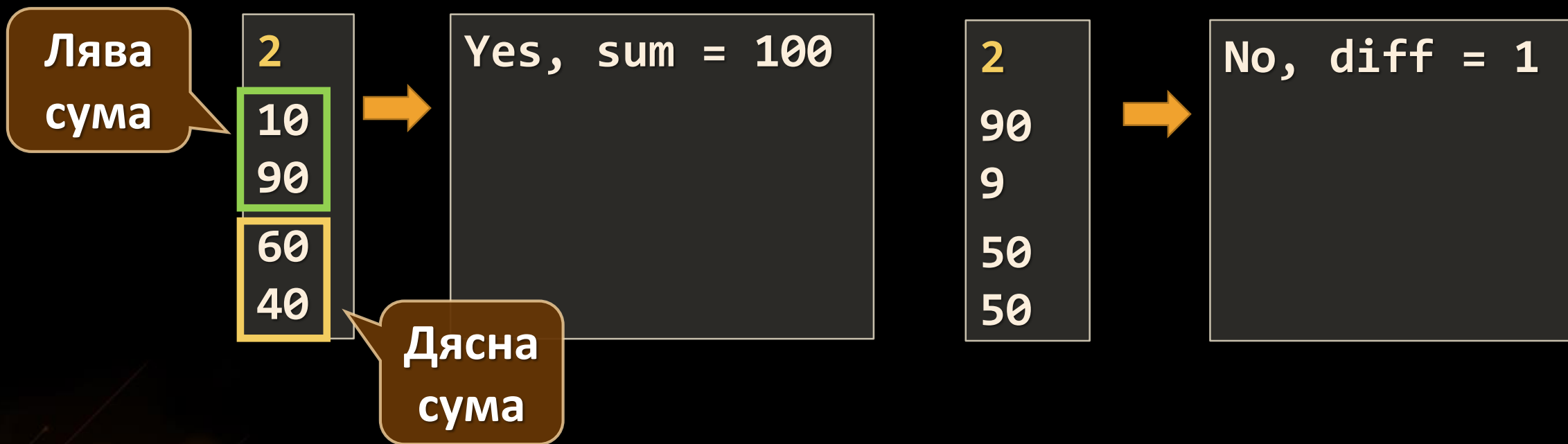
Лява и дясна сума - условие

- 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)

1 2 3 2 1 3

Лява и дясна сума - условие

- Примерен вход и изход:



Решение: лява и дясна сума

```
import math
n = int(input())
leftSum = 0
rightSum = 0
for i in range(0, n):
    numbersForLeftSide = float(input())
    leftSum += numbersForLeftSide
# Read the right part of the numbers and sum them
if (leftSum == rightSum):
    print("Yes, sum = %d" % leftSum)
else:
    print("No, diff = %d" % math.fabs(rightSum - leftSum))
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#6>

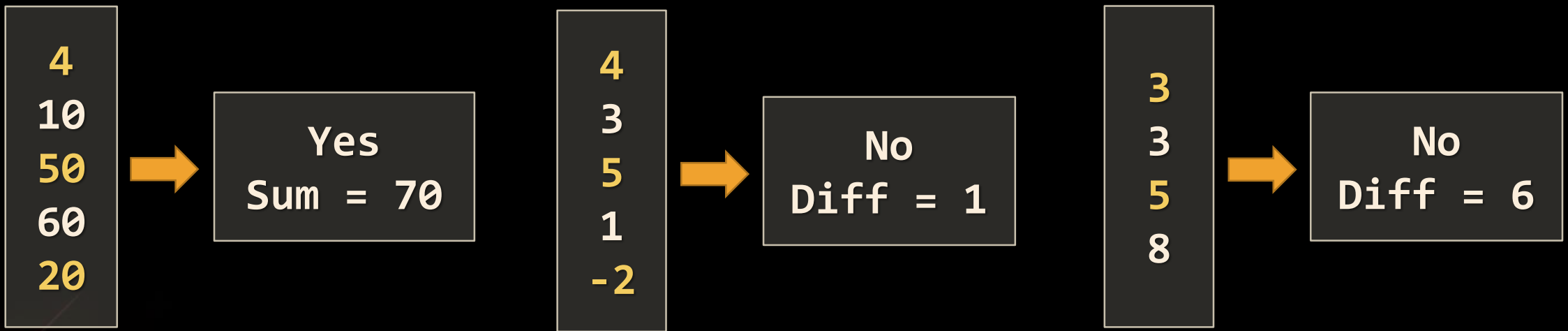
Четна / нечетна сума - условие

- 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**" + **сумата** иначе печата "**No**" + **разликата** (положително число).

3 5 3 1 2 2

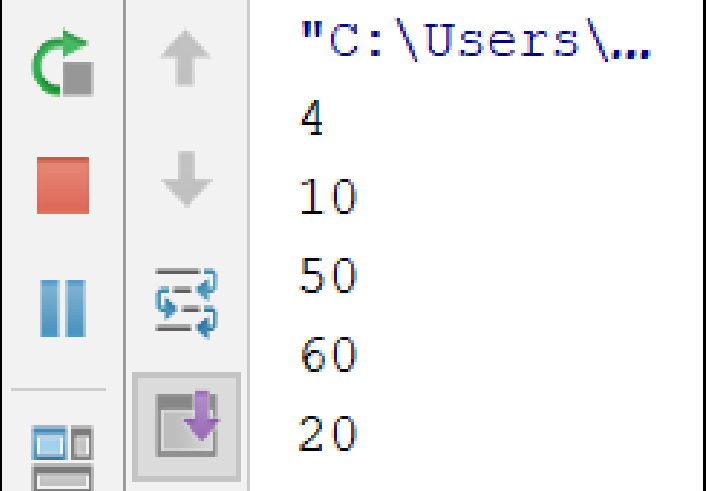
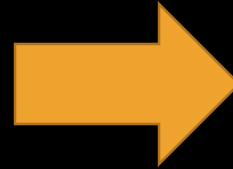
Четна / нечетна сума – условие (2)

- Примерен вход и изход:



Решение: четна / нечетна сума

```
n = int(input())
evenSum = 0
oddSum = 0
for i in range(1, n+1):
    num = int(input())
    if i % 2 == 0:
        evenSum += num
    else:
        oddSum += num
# Print sum or difference
```



```
"C:\Users\...
4
10
50
60
20
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#7>

Сумиране на гласните букви - условие

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

а	е	и	о	и
1	2	3	4	5

- Примерен вход и изход:

hello → 6 ($e+o = 2+4 = 6$)

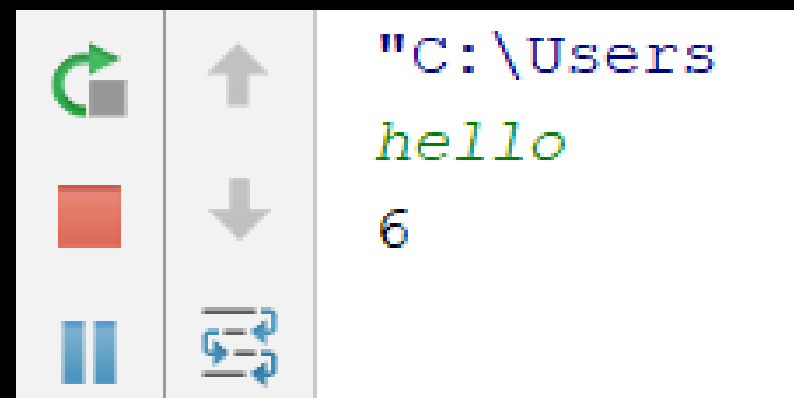
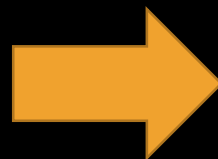
hi → 3 ($i = 3$)

bamboo → 9 ($a+o+o = 1+4+4 = 9$)

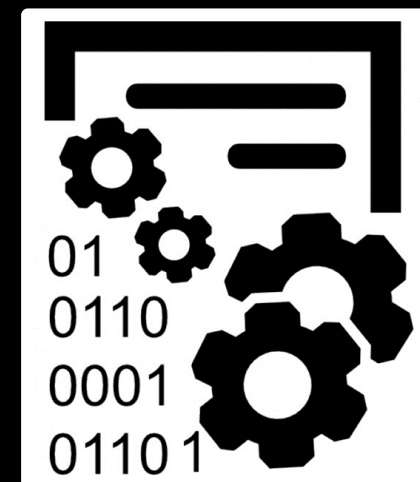
beer → 4 ($e+e = 2+2 = 4$)

Сумиране на гласни букви - решение

```
str = input().lower()
sum = 0
for c in str:
    if c == 'a':
        sum += 1
    elif c == 'e':
        sum += 2
# Check other vowels
print(sum)
```



Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/154#8>



Задачи с цикли

Работа на живо в клас (лаб)

Какво научихме днес?

- Можем да повтаряме блок код с **for**-цикъл:

```
for i in range(1, 101):  
    print('i = ' + str(i))
```

- Можем да четем поредица от **n** числа от конзолата:

```
for i in range(n):  
    num = int(input())
```

- Можем да преобразуваме данни с кастване:

```
a = (chr)(67)    # C  
b = ord('#')     # 35
```

Полученият резултат в **int** е
ASCII стойността на символа



Повторения (цикли)



Questions?

- Настоящият курс (слайдове, примери, видео, задачи и др.) се разпространяват под свободен лиценз "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International"



- Благодарности: настоящият материал може да съдържа части от следните източници
 - Книга "Основи на програмирането със C#" от Светлин Наков и колектив с лиценз CC-BY-SA

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education, Profession and Job for Software Developers
 - softuni.bg
- Software University Foundation
 - <http://softuni.foundation/>
- Software University @ Facebook
 - facebook.com/SoftwareUniversity
- Software University Forums
 - forum.softuni.bg



**Software
University**

