

# Преговор

1

## Преговор



#### 1. Колко пъти ще се изпише "SoftUni" на конзолата след изпълнението на следния код:

int i = 0; 0

while(i <= 5) { 5

System.out.println("SoftUni");

i++;

**4**

}

### 6

**Преговор**



#### Колко пъти ще се изпише "SoftUni" на конзолата след изпълнението на следния код:

int i = 0; 0

while(i <= 5) { 5

System.out.println("SoftUni");

i++;

### 4

}

### 6

**Преговор**

#### Колко пъти ще се изпише "SoftUni" на конзолата след изпълнението на следния код:

int i = 0;



while(i == 0) { System.out.println("SoftUni"); if(i == 1)

break;

}

### 0

Безброй много пъти

**1**

### 100000

**Преговор**

#### Колко пъти ще се изпише "SoftUni" на конзолата след изпълнението на следния код:

int i = 0;



while(i == 0) { System.out.println("SoftUni"); if(i == 1)

break;

}

### 0

Безброй много пъти

**1**

### 100000

**Преговор**

#### Колко итерации ще има следния цикъл:

int i = 0; while(i <= 10) {

i--;

**Безброй**

**10 много**

}



## 11

### 2147483647

**Преговор**

#### Колко итерации ще има следния цикъл:

int i = 0; while(i <= 10) {

i--;

**Безброй**

**10 много**

}



## 11

### 2147483647

**Преговор**

#### Какъв ще е резултатът от изпълнението на следния код:

int i = 0; while (i <

i++;

6) {

**024**

## 24

if (i % 2 == 0) System.out.print(i);



}

**246**

**123456**

**Преговор**

4. Какъв ще е резултатът от изпълнението на следния код:

int i = 0; while (i <

i++;

6) {

**024**

## 24

if (i % 2 == 0) System.out.print(i);



}

**246**

**123456**

**Преговор**

5. Какъв ще е резултатът от изпълнението на следния код:

char a = 'a'; while (a < 100) {

System.out.print(a);

a++;

**Безкраен**

**цикъл**

**abc**



}

**ааа Грешка при**

**компилация**

**Преговор**

5. Какъв ще е резултатът от изпълнението на следния код:

char a = 'a'; while (a < 100) {

System.out.print(a);

a++;

**Безкраен**

**цикъл**

**abc**



}

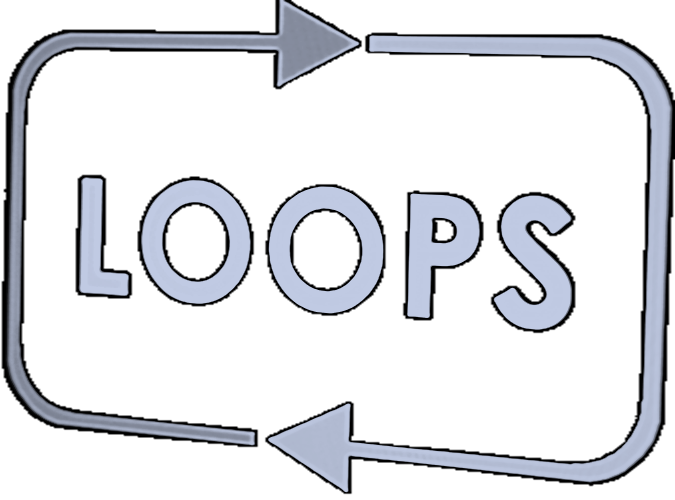
**ааа Грешка при**

**компилация**

**Повторения (цикли)**

Прости повторения с for-цикъл

СофтУни



Преподавателски екип

**Софтуерен университет**

[**http://softuni.bg**](http://softuni.bg/)



**Имате въпроси?**

**sli.do**

**#pb-oct**

13



# Повторения на блокове код

**Конструкция за for-цикъл**

**for-цикъл - конструкция**

* Можем да повтаряме действия до определен момент чрез

**for**-цикли

Ключова дума за конструкцията

**Начална стойност**

**Крайна стойност**

**for (int i = 1; i <= 10; i++) {**

**System.out.print(i);**

**}**

**Инкрементация на индекса (i)**

**Тяло на цикъла: блок от код за**



**повторение**

16

## Числа от 1 до 100

###### Напишете програма, която:

* + Извежда числата [1…100] всяко на нов ред

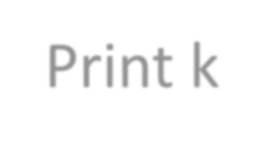
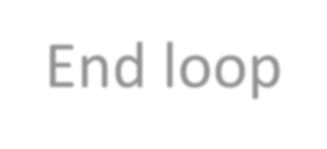
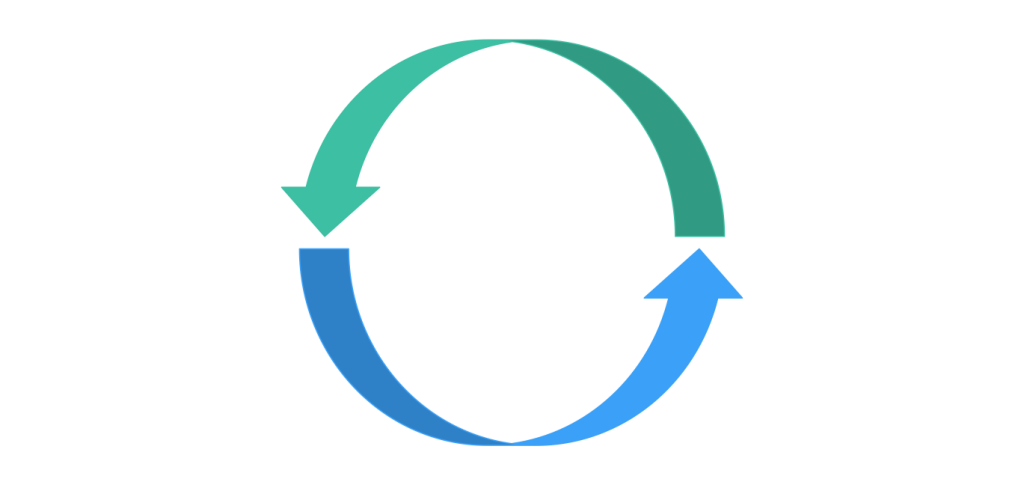
i = 1

i <=

100 false End loop

Принтиране

true



Print k

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#0> 17

## ASCII таблица



#### Символите, които използваме се представят като числа

###### Поместени са в ASCII таблицата

* Примери (знак и неговата ASCII стойност):

A

**97**

**65**

**a**

**64 + 43**

**@**

[Пълна информация за ASCII](https://bg.wikipedia.org/wiki/ASCII) 18

## Unicode таблица



#### Съвременен вариант на ASCII таблицата

###### Съдържа 137 439 знака и обхваща 146 писмености

* Примери (знак и неговата Unicode стойност):

'\u00A2'

**'**

**'\u005A**

**Z**

**¢**

**'\u00B5'**

**µ**

**'\u007E'**

**~**

[Пълна информация за Unicode](https://unicode-table.com/en/) 19

## Преобразуване на типове данни

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

* Примери:
  + От реално към цяло число:

Получава се загуба на данни

**(десетичната част от числото)**

**int a = int b =**

**(int)5.66; // 5**

**(int)5.44; // 5**

* + От символ към число и обратно: **char a = (char)67; // C int b = '#'; // 35**



Полученият резултат в int е

**ASCII стойността на символа**

20

## Всички латински букви - условие

#### Напишете програма, която:

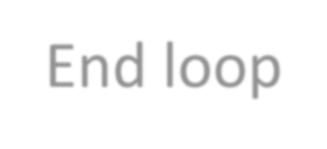
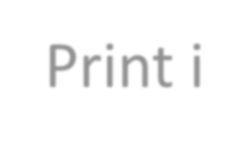
###### Извежда буквите от латинската азбука: **[a, z]**

i = 'a'

i <= 'z'

false

End loop



true

Print i

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#1> 21

## Работа с текст



#### Можем да вземем дължината на текст

String text = scanner.nextLine(); // въвеждаме SoftUni int length = text.length(); // 7

* Можем да вземем символ от текст по индекс

**String text = scanner.nextLine(); // въвеждаме SoftUni char letter = text.charAt(4); // U**

22

## Сумиране на гласните букви - условие

#### Напишете програма, която:

###### Прочита от потребителя текст

* + Извежда сумата на гласните букви според таблицата по-долу:

**a e i o u 1 2 3 4 5**

* Примерен вход и изход:

**beer**

**hi**

hello

**6** (e+o = 2+4

= 6)

**3** (i = 3)

bamboo



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

**4** (e+e = 2+2 = 4)

23

## Сумиране на гласни букви - решение

Scanner scanner = new Scanner(System.in);

**String input = scanner.nextLine();**

**int sum = 0;**

**for (int i = 0; i <**

**Можем да вземем дължината на текста**

**input.length(); i++) {**

**switch (input.charAt(i)) case 'a': sum += 1; break; case 'e': sum += 2; break;**

**Можем да вземем символ по индекс i**

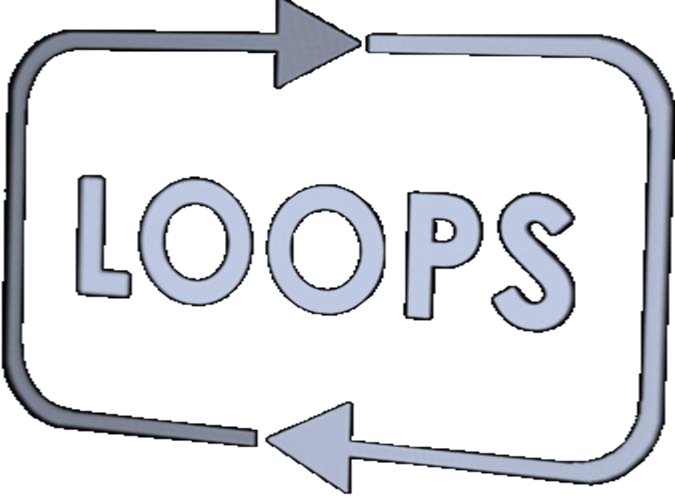
**// TODO: Add cases for other vowels.**



**}**

**System.out.println("Vowels sum = " + sum);**

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#3> 24



# Повторения на блокове код

**Решаване на задачи в клас (лаб)**



**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:

**4**

**3**

**45**

**-20**

**7**

**11**

**2**

**10**

**20**

**-10**

**43**

**-60**

**30**

**-20**

**-30**

27

Read input

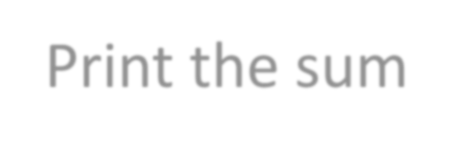
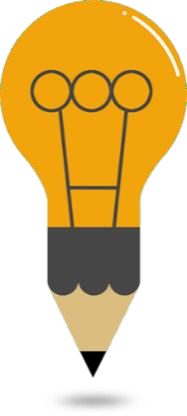
i = 1

sum = 0

i <= n

false

Print the sum



true

Принтиране на сумата

Read a number; Add it to the sum

## Редица цели числа - условие



###### Напишете програма, която:

* + Чете **n** на брой цели числа
  + Принтира най-голямото и най-малкото число

Max number: 304

5

10

20

304

0

50

Min number: 0

29

smallest = int.MaxValue

biggest = int.MinValue

Read n

i = 0

i < n

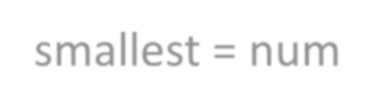
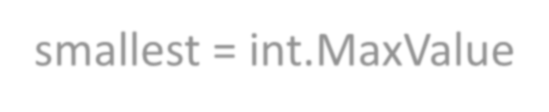
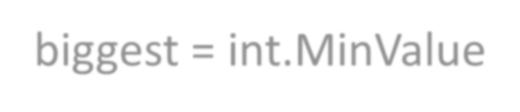
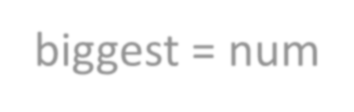
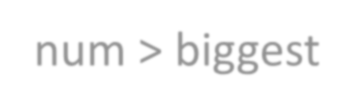
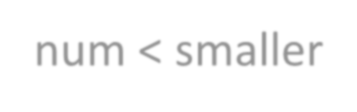
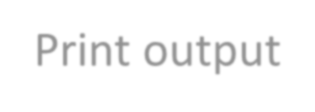
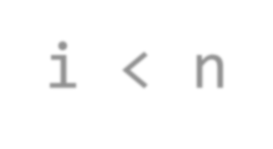
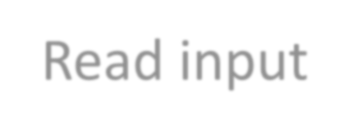
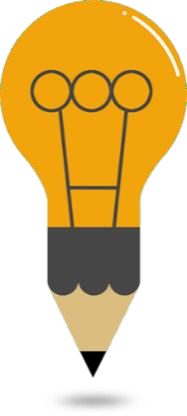
false

Print output

true

biggest = num

num > biggest



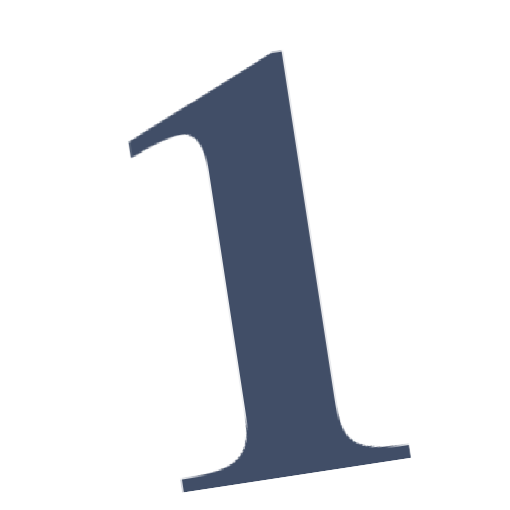
Read input

num < smaller

smallest = num

30

## Редица цели числа - решение



int smallest = Integer.MAX\_VALUE; int biggest = Integer.MIN\_VALUE;

int n = Integer.parseInt(scanner.nextLine());

for (int i = 0; i < n; i++) {

int num = Integer.parseInt(scanner.nextLine());

if (num < smallest) smallest = num; if (num > biggest) biggest = num;

}

System.out.printf("Max number: %d%n", biggest);

System.out.printf("Min number: %d", smallest);

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#5>

31

## 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)

32

## Лява и дясна сума - условие

###### Примерен вход и изход:

Лява сума

**2**

**90**

**10**

**90**

**2**

**Yes, sum = 100 9**

**No, diff = 1**

**60**

**40**

**50**

**50**

**Дясна**

**сума**



33

## Решение: лява и дясна сума

Scanner int n =

scanner = new Scanner(System.in); Integer.parseInt(scanner.nextLine());

int leftSum = 0;



for (int i = 1; i <= n; i++) {

leftSum += Integer.parseInt(scanner.nextLine());

}

// TODO: read and calculate the rightSum if (leftSum == rightSum)

System.out.println("Yes, sum = " + leftSum);

else

int diff = Math.Abs(rightSum - leftSum); System.out.println("No, diff = " + diff);

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#6> 34

## 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).

35

## Четна / нечетна сума - условие

* Примерен вход и изход:

**4**

**4**

**10**

**50**

**60**

**20**

**Yes 3**

**Sum = 70 5**

**1**

**-2**

**3**

**5 No**

**No**

**Diff = 1**

**8 Diff = 2**

**1**



36

## Решение: четна / нечетна сума



**Scanner scanner = new Scanner(System.in);**

**int n = Integer.parseInt(scanner.nextLine()); int oddSum = 0;**

**int evenSum = 0;**

**for (int i = 1; i <= n; i++) {**

**int element = Integer.parseInt(scanner.nextLine()); if (i % 2 == 0) evenSum += element;**

**else oddSum += element;**

**}**

**// TODO: print the sum / difference**

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#7> 37



# По-сложни задачи с цикли

**Решаване на задачи в клас (лаб)**



**Работа с по-сложни for-цикли**

**Цикли със стъпка**

### 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:

**100, 99, 98, …, 3, 2, 1**

**100**

40

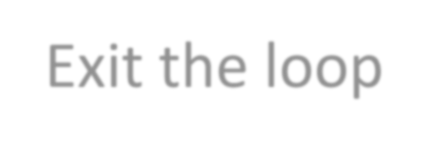
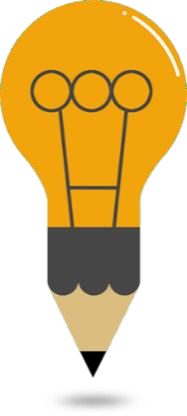
Read n

i = n

i >= 1

false

Exit the loop



true

print i;

i --;

41

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

##### Scanner scanner = new Scanner(System.in);

**int n = I~~n~~teger.parseInt(scanner.nextLine());**

**for (int**

**i = n;**

**i >= 1;**

**i--) {**

Намаляваща стъпка: -1

**System.out.println(i);**

**}**

**Обърнато условие: i >= 1**



Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#8> 42

## 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:

**1, 4, 7, 10**

**10**

43

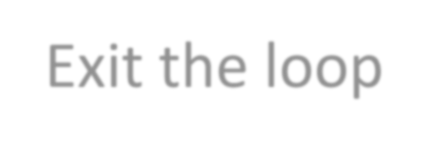
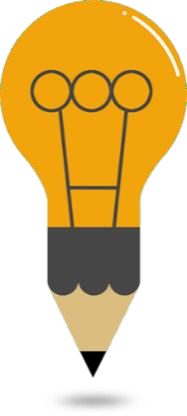
Read n

i = 1

i <= n

false

Exit the loop



true

print i;

i += 3;

44

## Числата от 1 до N през 3 – решение

##### Scanner

**int n =**

**scanner = new Scanner(System.in);**

**Integer.parseInt(scanner.nextLine());**

**for (int i = 1; i <= n; i += 3) {**

**System.out.println(i);**

**}**

Задаване на стъпка 3



Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#9> 45

## Четни степени на 2 – условие



#### Напишете програма, която:

###### Прочита цяло число **n**

* + Отпечатва четните степени на **2** до **2n**: 20, 22, 24, 26, …, **2n**
* Примерен вход и изход:

**1, 4, 16 , …, 1024**

**10**

**1, 4, 16 , …, 64**

**7**

46

## Четни степени на 2 – решение

##### Scanner scanner = new Scanner(System.in);

**int n = Integer.parseInt(scanner.nextLine());**

**int num = 1;**

**for (int i = 0; i <= n; i += 2) {**

**System.out.println(num);**

**num = num \* 2 \* 2;**

**}**

Ползваме стъпка 2



Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1015#10> 47

## Какво научихме днес?



* + Символите, които използваме се представят като

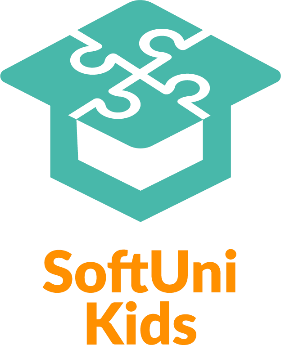
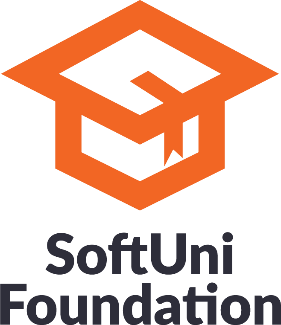
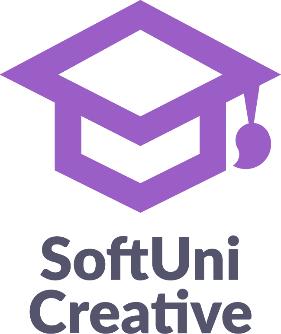
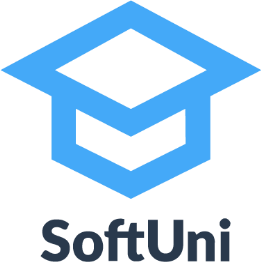
числа и са поместени в **ASCII** таблицата

* + Можем да повтаряме блок код с **for**-цикъл
  + Можем да преобразуваме типове от данни чрез

**кастване**

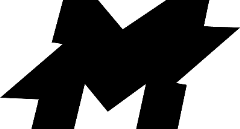
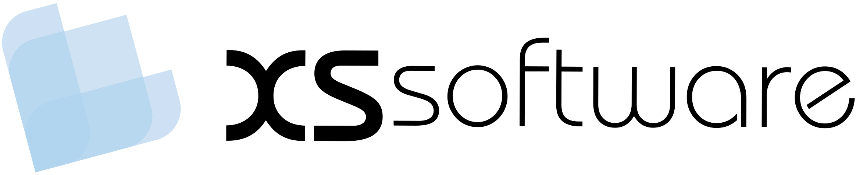
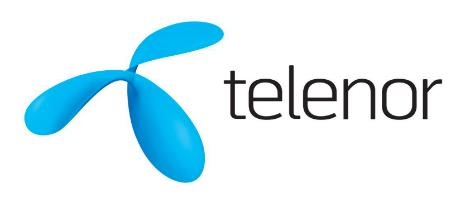
* + Символите могат да се репрезентират като числа
  + Можем да четем поредица от **n** числа от конзолата
  + Можем да вземем символ по индекс от текст
  + Можем да използваме **for**-цикли със стъпка

48



<https://softuni.bg/courses/programming-basics>

## SoftUni Diamond Partners



**SoftUni Organizational Partners**



**Лиценз**



###### Настоящият курс (слайдове, примери, видео, задачи и др.)

[се разпространяват под свободен лиценз "Creative Commons Attribution-~~N~~onCommercial-ShareAlike 4.0 International"](http://creativecommons.org/licenses/by-nc-sa/4.0/)

* Благодарности: настоящият материал може да съдържа части от следните източници
  + Книга "[Основи на програмирането със Java"](https://java-book.softuni.bg/) от Светлин Наков и колектив с лиценз [CC-BY-SA](http://creativecommons.org/licenses/by-sa/4.0/)

52

## Обучения в СофтУни



###### Software University – High-Quality Education, Profession and Job for Software Developers

* + [softuni.bg](http://softuni.bg/)
* Software University Foundation
  + <http://softuni.foundation/>
* Software University @ Facebook
  + [facebook.com/SoftwareUniversity](https://www.facebook.com/SoftwareUniversity)
* Software University Forums
  + [forum.softuni.bg](http://forum.softuni.bg/)