



Преговор

1. Каква ще е стойността на променливата **a** след изпълнението на следната програма:

```
int a = 5;  
switch (a) {  
    case 5:  
    case 6:  
        a = a + 1;  
        break;  
    default:  
        a = a + 2;  
        break;  
}
```

0

5

6

7

1. Каква ще е стойността на променливата **a** след изпълнението на следната програма:

```
int a = 5;  
switch (a) {  
    case 5:  
    case 6:  
        a = a + 1;  
        break;  
    default:  
        a = a + 2;  
        break;  
}
```

0

5

6

7

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

```
System.out.println(!(5 == 5) && (4 + 1 == 5));
```

True

False

Runtime
error

Compile time
error

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

```
System.out.println(!(5 == 5) && (4 + 1 == 5));
```

True

False

Runtime
error

Compile time
error

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

```
System.out.println(!(3 == 3) || (3 == 5));
```

True

Runtime
error

False

Compile time
error

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

```
System.out.println(!(3 == 3) || (3 == 5));
```

True

Runtime
error

False

Compile time
error

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

```
System.out.println(!(3 > 5) || (1 == 1));
```

Compile time
error

Runtime
error

False

True

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

```
System.out.println(!(3 > 5) || (1 == 1));
```

Compile time
error

Runtime
error

False

True

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

```
int number = 101;  
if (number >= 1)  
    System.out.println("Larger than 1");  
if (number <= 101)  
    System.out.println("Less than 101");  
    System.out.println("Equal to 101");
```

Larger than 1

Less than 101

Equal to 101

Compile
time error

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

```
int number = 101;  
if (number >= 1)  
    System.out.println("Larger than 1");  
if (number <= 101)  
    System.out.println("Less than 101");  
    System.out.println("Equal to 101");
```

Larger than 1

Less than 101

Equal to 101

Compile
time error

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

```
String role = "Administrator";  
String password = "SoftUni";  
if(role.equals("SoftUni")) {  
    if(password.equals("SoftUni")) {  
        System.out.println("Welcome!");  
    }  
}
```

"Welcome!"

Runtime error

No output

Compile time
error

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

```
String role = "Administrator";  
String password = "SoftUni";  
if(role.equals("SoftUni")) {  
    if(password.equals("SoftUni")) {  
        System.out.println("Welcome!");  
    }  
}
```

"Welcome!"

Runtime error

No output

Compile time
error

Повторения (цикли)

Прости повторения с `while`-цикъл



СофтУни

Преподавателски екип



Software
University



SoftUni
Foundation



Софтуерен университет
<http://softuni.bg>

Имате въпроси?

sli.do

#pb-nov



**Увеличаване и намаляване на стойността
на променливи**

- Инкрементиране - увеличаването на стойността на дадена променлива
 - Извършва се чрез оператори за инкрементиране - префиксни и постфиксни

Пример	Име	Резултат
<code>++a</code>	Пре-инкрементация	Увеличава стойността с единица и връща <code>a</code>
<code>a++</code>	Пост-инкрементация	Връща <code>a</code> и увеличава стойността с единица

- Извършва се само върху променливи, които имат числена стойност

- Пре-инкрементация

```
int a = 1;  
System.out.println(++a); // 2  
System.out.println(a);   // 2
```

Стойността на променливата a се увеличава с 1 и след това се принтира

- Пост-инкрементация

```
int a = 1;  
System.out.println(a++); // 1  
System.out.println(a);   // 2
```

Първо се принтира променливата a и след това се увеличава с 1

- Декрементиране – намаляването на стойността на дадена променлива
 - Извършва се чрез оператори за декрементиране – префиксни и постфиксни

Пример	Име	Резултат
--a	Пре-декрементация	Намалява стойността с единица и връща a
a--	Пост-декрементация	Връща a и намалява стойността с единица

- Извършва се само върху променливи, които имат числена стойност

- Пре-декрементация

```
int a = 1;  
System.out.println(--a); // 0  
System.out.println(a);  // 0
```

Стойността на променливата a се намалява с 1 и след това се принтира

- Пост-декрементация

```
int a = 1;  
System.out.println(a--); // 1  
System.out.println(a);  // 0
```

Първо се принтира променливата a и след това се намалява с 1

A background network diagram consisting of a grid of light gray lines intersecting at various points. At these intersections, there are several circles of different sizes, some of which are filled with a light gray color, while others are empty. The overall pattern resembles a molecular structure or a network graph.

while

while-цикъл

Повторение докато е вярно дадено условие

Повторения (цикли) – while-цикъл

- В програмирането често се налага да изпълним блок с команди няколко пъти
 - За целта използваме цикли – **while**, **for** и други



while-цикъл – пример

```
int a = 5;
```

Условие за прекратяване
на повторението

```
while (a <= 10) {  
    System.out.println("a = " + a);  
    a++;  
}
```



a = 5

a = 6

a = 7

a = 8

a = 9

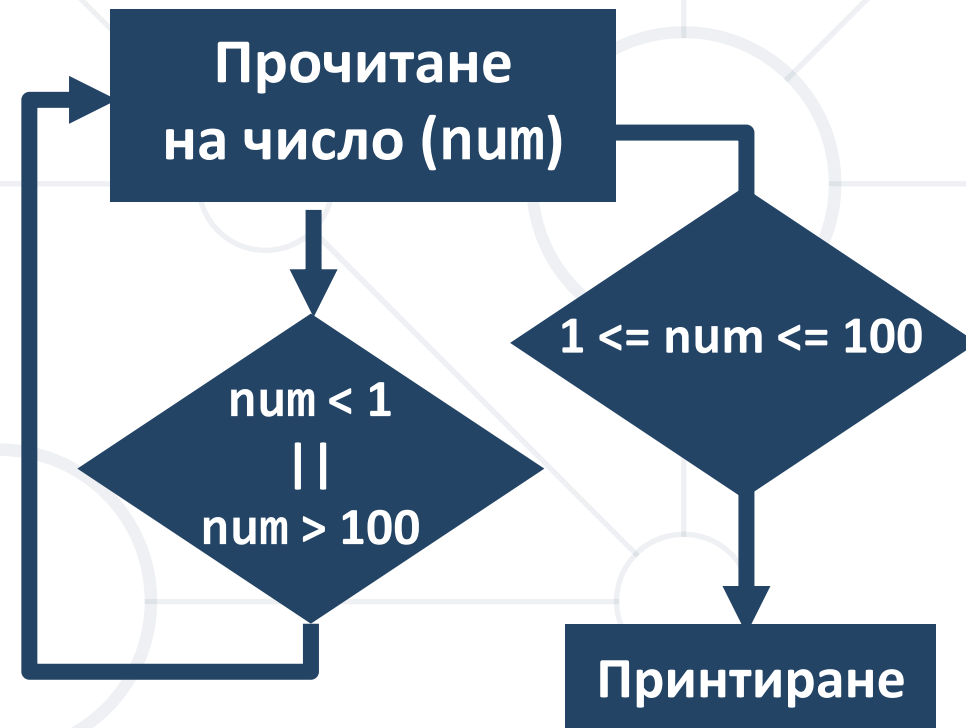
a = 10

Press any key to continue . . .



Число в диапазона [1...100] - условие

- Напишете програма, която:
 - Прочита цяло число
 - Проверява дали е в диапазона [1...100]
- При:
 - Невалидно число, прочита ново
 - Намиране на число в диапазона, прекратява изпълнение



Число в диапазона [1...100] - решение

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

while (num < 1 || num > 100) {
    System.out.println("Invalid number!");
    num = Integer.parseInt(scanner.nextLine());
}

System.out.println("The number is: " + num);
```

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1014#0>

- Напишете програма, която:
 - Първоначално прочита потребителско име и парола на потребителски профил
 - Прочита парола за вход и проверява дали е коректна
 - При:
 - Невалидна парола, прочита нова
 - При коректно въведена парола, прекратява изпълнение

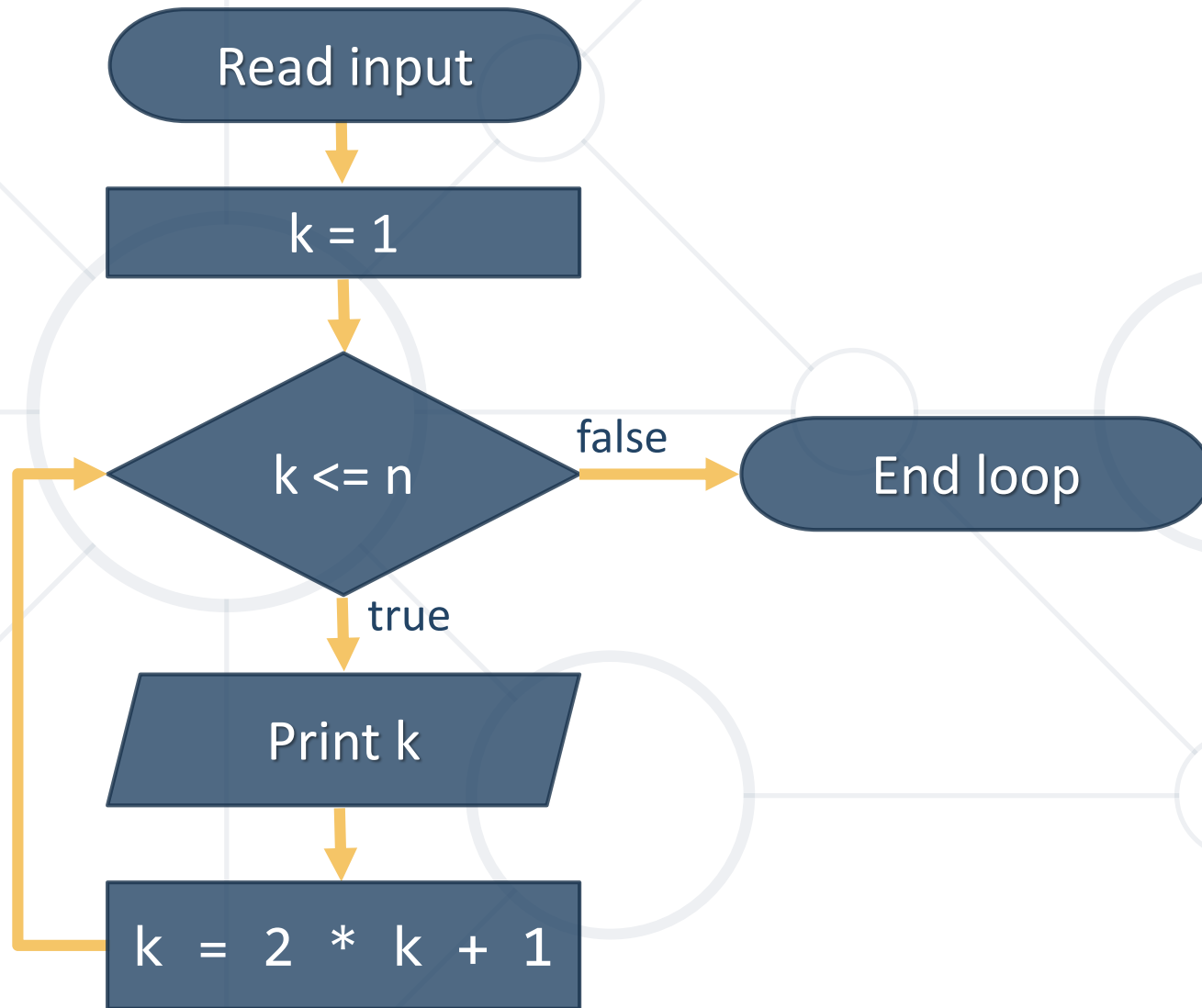
```
Scanner scanner = new Scanner(System.in);  
String username = scanner.nextLine();  
String password = scanner.nextLine();  
  
String input = scanner.nextLine();  
while (!input.equals(password)) {  
    input = scanner.nextLine();  
}  
  
System.out.printf("Welcome: %s!", username);
```

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1014#1>

Редица числа $2k+1$ - условие

- Напишете програма, която:
 - Прочита цяло число **n**
 - Отпечатва всички числа $\leq n$ от редицата: 1, 3, 7, 15, 31, ...
 - Всяко следващо число е равно на **предишното** $\ast 2 + 1$

1, $(1 \ast 2) + 1 =$ **3**, $(3 \ast 2) + 1 =$ **7**, $(7 \ast 2) + 1 =$ **15** ...



Редица числа $2k+1$ - решение

```
int n = Integer.parseInt(scanner.nextLine());
int k = 1;
while (k <= n) {
    System.out.println(k);
    k = 2 * k + 1;
}
```

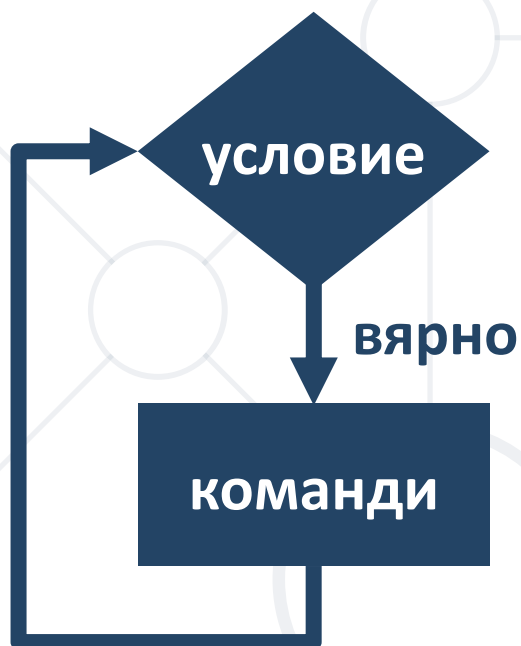
Повторение докато е в
сила условието $k \leq n$



Прекъсване чрез оператор `break`

Безкрайни цикли

- Безкраен цикъл – повтаряне на блок от код безкраен брой пъти:



Условието е винаги
вярно

```
while (true) {  
    System.out.println("Infinite loop");  
}
```



- Оператор **break** – прекъсва цикъла

```
while (true) {  
    System.out.println("Infinite loop");  
    if (...) {  
        break;  
    }  
}
```

Условие за прекъсване на
цикъла

- Напишете програма, която:
 - Чете n – на брой числа, които представляват вноски по банкова сметка
 - При всяка вноска принтира:
"Increase: {сумата}"



Баланс на сметка - условие (2)

- Ако се въведе отрицателно число да се изпише **"Invalid operation!"** и програмата да приключи
- Накрая на програмата трябва да се изпише:
"Total: {общата сума в сметката}"



Баланс на сметка - условие(3)

- Примерен вход и изход:

3
5.51
69.42
100



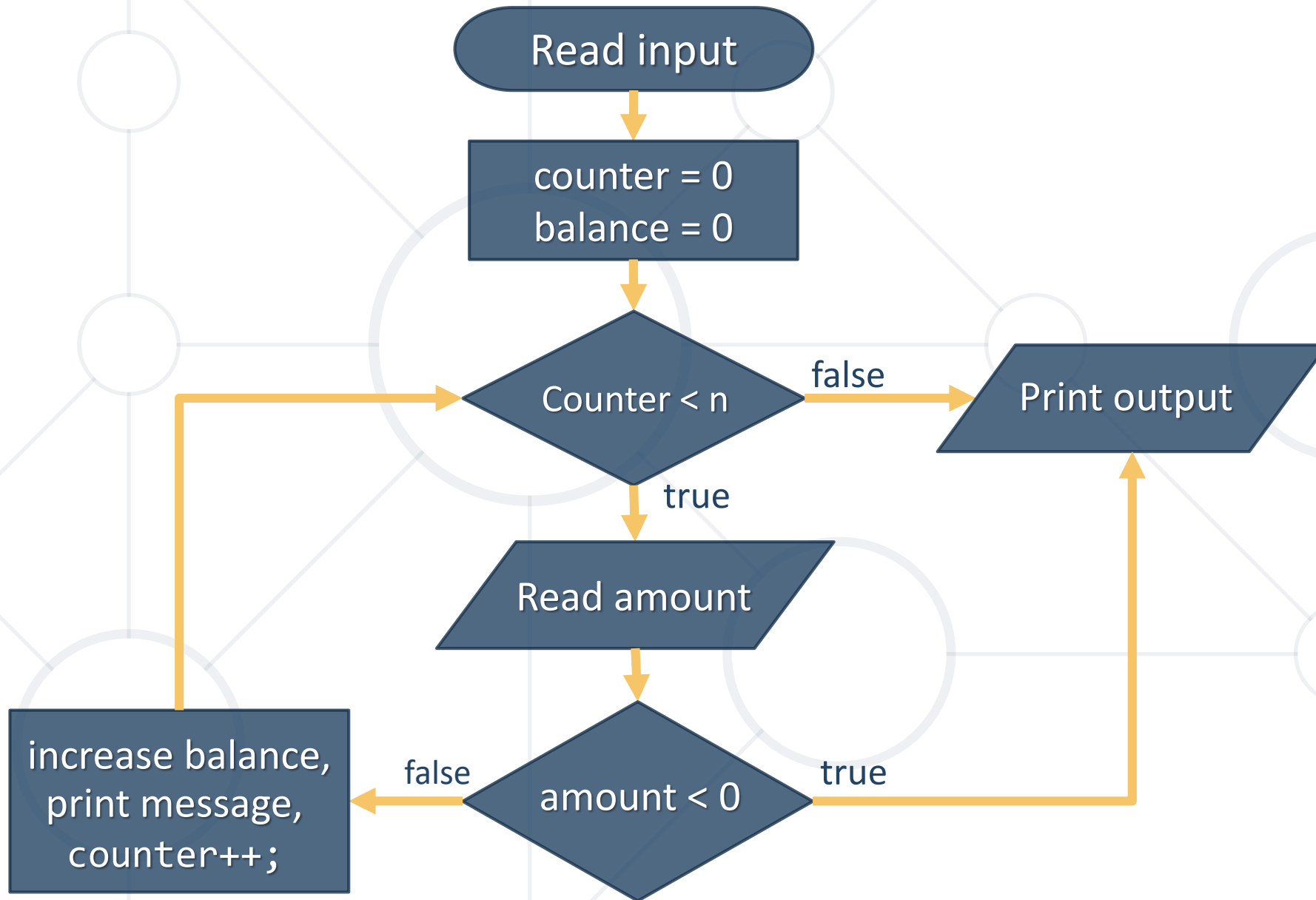
Increase: 5.51
Increase: 69.42
Increase: 100.00
Total: 174.93

5
120
45.55
-150



Increase: 120
Increase: 45.55
Invalid operation!
Total: 165.55

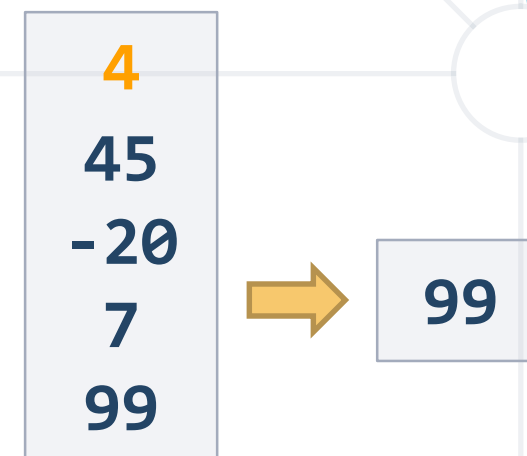
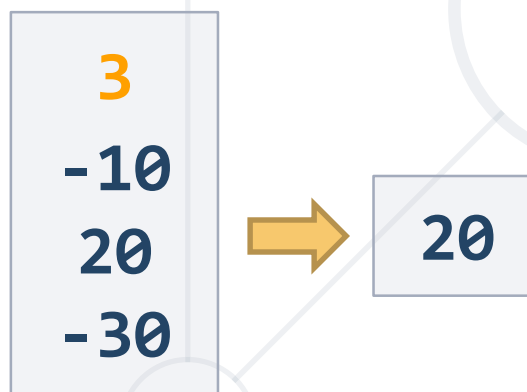
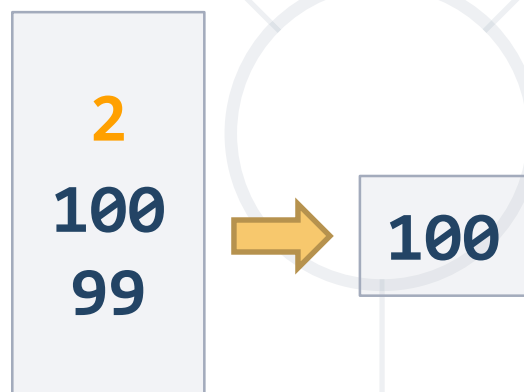




```
int n = Integer.parseInt(scanner.nextLine());
int counter = 0;
double balance = 0.0;
while (counter < n) {
    double amount = Double.parseDouble(scanner.nextLine());
    if (amount < 0) { //TODO: Print output and exit the loop
        balance += amount;
        System.out.printf("Increase: %.2f", amount);
        counter++;
    }
    System.out.printf("Total: %.2f", balance);
}
```

Най-голямо число - пример

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



Най-голямо число - решение

```
Scanner scanner = new Scanner(System.in);
int n = Integer.parseInt(scanner.nextLine());
int count = 0;
int max = Integer.MIN_VALUE;
while (count < n) {
    int num = Integer.parseInt(scanner.nextLine());
    count++;
    if (num > max) {
        max = num;
    }
}
System.out.println(max);
```


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:

2
99
100



99

3
-10
20
-30



-30

4
45
-20
7
99



-20



Най-малко число - решение

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

int n = Integer.parseInt(scanner.nextLine());
int count = 0;
int min = Integer.MAX_VALUE;

while (count < n) {
    //TODO: Use logic similar to the previous problem
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1014#5>

- Напишете програма, която:
 - Изчислява **средната оценка** на ученик от цялото му обучение
 - Ако годишната му оценка е:
 - ≥ 4.00 , ученикът преминава е следващия клас
 - < 4.00 , той ще повтори класа
 - При **завършване** да се отпечата:
"{име на ученика} graduated. Average grade: {средната оценка от цялото обучение}"

Завършване - условие (2)

- Примерен вход и изход:

Pesho
4
5.5
6
5.43
4.5
6
5.55
5
6
6
5.43
5



Pesho graduated.
Average grade: 5.37

Ani
5
5.32
6
5.43
5
6
5.5
4.55
5
6
5.56
6



Ani graduated.
Average grade: 5.45

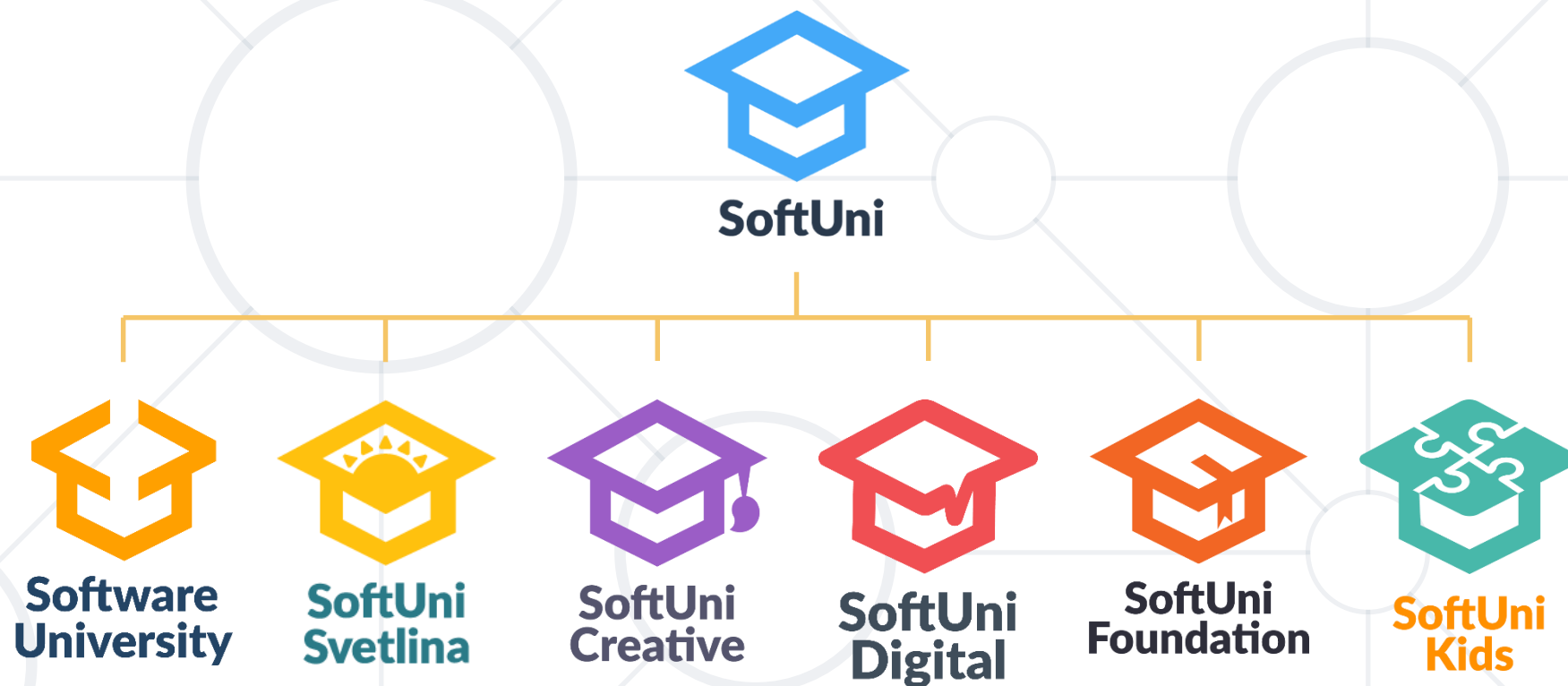
```
String name = scanner.nextLine();
int counter = 1;
double sum = 0;
while (counter <= 12) {
    double grade = Double.parseDouble(scanner.nextLine());
    if (grade >= 4.00) {
        sum += grade;
        counter++;
    }
}
double average = sum / 12;
//TODO: print the output
```



- Можем да инкрементираме/декрементираме числови стойности
- Използваме **while** - цикли, за да повтаряме действие, докато е в сила дадено условие
- Можем да прекъсваме циклите с оператора **break**



Въпроси?



SoftUni Diamond Partners



XSsoftware



SBTech
we know sports



telenor



SoftwareGroup
doing it right

NETPEAK



SmartIT



Postbank

Решения за твоето утре



INDEAVR

Serving the high achievers



INFRAGISTICS®



STEMO®
Computer Systems & Software

SUPERHOSTING.BG

SoftUni Organizational Partners

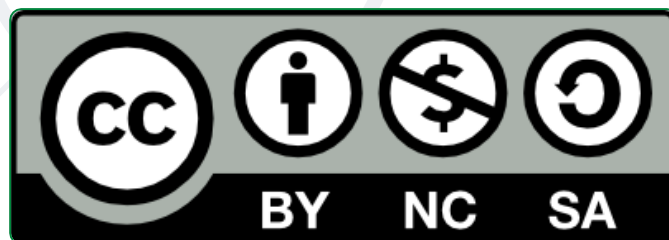


OneBit
SOFTWARE



WORLD
OF
MYTHS

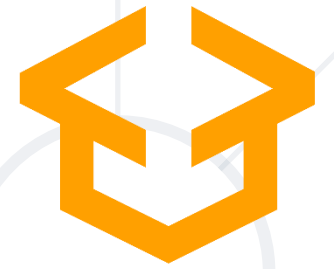
- Настоящият курс (слайдове, примери, видео, задачи и др.) се разпространяват под свободен лиценз "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International"



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

Обучения в СофтУни

- 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](https://www.facebook.com/SoftwareUniversity)
- Software University Forums
 - forum.softuni.bg



**Software
University**

