# Упражнения: Повторения (цикли)

Тествайте решенията си в **Judge системата**: <https://judge.softuni.bg/Contests/3157/Loops>.

TODO: да се внимава насоките да са най-подробни в първите задачи, да намаляват в следващите няколко задачи и да изчезват напълно в последните задачи.

TODO: в упражненията от следващите 2 теми съм отбелязал няколко по-лесни задачи, които да дойдат в това упражнения, за да станат 14-15 задачите тук.

## Числа от 1 до 100

Напишете програма, която отпечатва числата от 1 до 100, всяко на нов ред.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| (няма) | 1  2  3  …  98  99  100 |

### Насоки

1. Създайте **нов проект с име** "Numbers1To100".
2. Отидете в тялото на метода Main(String[] args) и напишете решението на задачата. Можете да си помогнете с кода от картинката по-долу:

Graphical user interface, text

Description automatically generated

## Числата от N до 1 в обратен ред

Напишете програма, която чете цяло положително число n, въведено от потребителя, и печата **числата от** n **до** **1** **в обратен ред** (от най-голямото към най-малкото).

### Примери

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |
| 2 | 2  1 | 3 | 3  2  1 | 5 | 5  4  3  2  1 |

### Насоки

1. Създайте **нов проект с име** "NumbersNTo1".
2. Отидете в тялото на метода Main(String[] args) и напишете решението на задачата. Можете да си помогнете с кода от картинката по-долу и обърнете внимание, че се използва цикъл с отрицателна стъпка:

Text

Description automatically generated

## Числата от 1 до N през 3

Напишете програма, която чете число n, въведено от потребителя, и отпечатва **числата от 1 до** n **през 3** (със стъпка 3).

### Примери

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |
| 10 | 1  4  7  10 | 7 | 1  4  7 | 15 | 1  4  7  10  13 |

### Насоки

1. Създайте **нов проект с име** "Number1ToNWithStep3".
2. Отидете в тялото на метода Main(String[] args) и напишете решението на задачата. Можете да си помогнете с кода от картинката по-долу и обърнете внимание на цикъла, че е със стъпка 3:

TODO: да се замъгли част от кода.

Graphical user interface, text

Description automatically generated

## Поток от символи

Напишете програма, която чете текст (стринг), въведен от потребителя, и печата всеки символ от текста на отделен ред.

### Примери

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Вход** | **Изход** |
| softuni | s  o  f  t  u  n  i | ice cream | i  c  e    c  r  e  a  m |

### Насоки

1. Прочетете входният текст:



1. Направете for цикъл с начална стойност на контролната променлива от 0 до input.Length (дължината на текста). На всяка итерация взимайте буквата на позиция във въведената дума равна на стойността на контролната променлива i.

TODO: да се замъгли част от кода.

A picture containing chart

Description automatically generated

1. На всяка итерация принтирайте стойността на променливата **letter**:

TODO: да се замъгли част от кода.

Text

Description automatically generated with medium confidence

## Сумиране на числа

Да се напише програма, която **чете** n на брой **цели числа**, въведени от потребителя, **и ги сумира**.

* От първия ред на входа се въвежда броят числа n.
* От следващите n реда се въвежда по едно цяло число.

Програмата трябва да прочете числата, да ги сумира и да отпечата сумата им.

### Примери

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |  | **Вход** | **Изход** |
| 2  10  20 | 30 | 3  -10  -20  -30 | -60 | 4  45  -20  7  11 | 43 | 1  999 | 999 | 0 | 0 |

## Четене на думи

Напишете програма, която чете текст от конзолата (стринг) и го принтира, докато не получи командата "**Stop**".

### Примери

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |
| Nakov  SoftUni  Sofia  Bulgaria  SomeText  Stop  AfterStop  Europe  HelloWorld | Nakov  SoftUni  Sofia  Bulgaria  SomeText |  | Sofia  Berlin  Moscow Athens  Madrid  London  Paris  Stop  AfterStop | Sofia  Berlin  Moscow Athens  Madrid  London  Paris |

## Парола

Напишете програма, която първоначално прочита име и парола на потребителски профил. След това чете парола за вход, при въвеждане на грешна парола, потребителя да се подкани да въведе нова парола.

### Примери

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вход** | **Изход** |  | **Вход** | **Изход** |
| Nakov  1234  pass  1324  1234 | Welcome Nakov! | Gosho  secret  secret | Welcome Gosho! |

### Насоки

1. Инициализирайте две променливи username и password, които ще съдържат потребителското име и паролата:

Graphical user interface

Description automatically generated

1. Инициализирайте променлива input, която ще държи въведената от потребителя парола за вход:



1. В while цикъл, до въвеждане на валидна парола, четете нова:

A picture containing text

Description automatically generated

1. Когато се въведе **валидна парола** **принтирайте съобщението за успешен вход**:

Graphical user interface, text, application

Description automatically generated

## Часовник

Напишете програма, която отпечатва **часовете в денонощието от 0:0 до 23:59**, всеки на отделен ред. Часовете трябва да се изписват във формат **"{час}:{минути}"**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| *(няма вход)* | 0:0  0:1  0:2  0:3  0:4  0:5  0:6  0:7  0:8  0:9  0:10  ...  23:50  23:51  23:52  23:53  23:54  23:55  23:56  23:57  23:58  23:59 |

### Насоки

1. Създайте 2 вложени for-цикъла, с които да итерирате през всяка една минута и час от денонощието:

Text

Description automatically generated with low confidence

1. Отпечатайте резултата в тялото на вътрешния цикъл.

## Таблица за умножение

Отпечатайте на конзолата таблицата за умножение за числата от 1 до 10 във формат:   
**"{първи множител} \* {втори множител} = {резултат}"**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| *(няма вход)* | 1 \* 1 = 1  1 \* 2 = 2  1 \* 3 = 3  1 \* 4 = 4  1 \* 5 = 5  1 \* 6 = 6  1 \* 7 = 7  1 \* 8 = 8  1 \* 9 = 9  1 \* 10 = 10  ...  10 \* 1 = 10  10 \* 2 = 20  10 \* 3 = 30  10 \* 4 = 40  10 \* 5 = 50  10 \* 6 = 60  10 \* 7 = 70  10 \* 8 = 80  10 \* 9 = 90  10 \* 10 = 100 |

### Насоки

1. Създайте 2 вложени for-цикъла, с които да итерирате всяка възможна стойност на двата множителя от 1 до 10:

Text, letter

Description automatically generated

1. Намерете произведението на двата множителя и отпечатайте резултата:

Text, letter

Description automatically generated