**Лаб: По-сложни цикли**

Задачи за упражнение и домашно към курса ["Основи на програмирането с Go".](https://softuni.bg/trainings/3683/programming-basics-with-go-february-2022)

**Тествайте** решенията си в **Judge системата**: <https://judge.softuni.org/Contests/3363/Advanced-Loops-Lab>

## Четене на думи

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

### **Примерен вход и изход**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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.
* при въвеждане на правилна парола: отпечатваме "Welcome {username}!".

### Примерен вход и изход

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |
| Nakov  1234  pass  1324  1234 | Welcome Nakov! | Gosho  secret  secret | Welcome Gosho! |

## Сума от числа

Напишете програма, която чете цяло число от конзолата и на всеки следващ ред цели числа, докато тяхната сума стане по-голяма или равна на първоначалното число. След приключване на четенето да се отпечата **сумата на въведените числа**.

### **Примерен вход и изход**

|  |  |  |  |
| --- | --- | --- | --- |
| **вход** | **изход** | **вход** | **изход** |
| 100  10  20  30  40 | 100 | 20  1  2  3  4  5  6 | 21 |

## Редица числа 2k + 1

Напишете програма, която чете число n, въведено от потребителя,и отпечатва **всички числа ≤** n **от редицата**: **1, 3, 7, 15, 31**, …. Всяко следващо число се изчислява като умножим **предишното** с **2** и добавим **1**.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 3 | 1  3 | 8 | 1  3  7 | 17 | 1  3  7  15 | 31 | 1  3  7  15  31 |

## Баланс по сметка

Напишете програма, която пресмята **колко общо пари** има в сметката, след като направите **определен брой вноски**. На всеки ред ще получавате сумата, която трябва да внесете в сметката, до получаване на команда "NoMoreMoney ". При всяка получена сума на конзолата трябва да се извежда **"Increase: "** + сумата и тя да се **прибавя в сметката**. Ако получите число **по-малко от 0** на конзолата трябва да се изведе **"Invalid operation!"** и **програмата да приключи**. Когато програмата приключи трябва да се принтира **"Total: "** + общата сума в сметката форматирана до втория знак след десетичната запетая.

### Примерен вход и изход

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |
| 5.51  69.42  100  NoMoreMoney | Increase: 5.51  Increase: 69.42  Increase: 100.00  Total: 174.93 | 120  45.55  -150 | Increase: 120.00  Increase: 45.55  Invalid operation!  Total: 165.55 |

## Най-голямо число

Напишете програма, която до получаване на командата "**Stop**", чете **цели числа**, въведени от потребителя и намира **най-голямото** измежду тях. Въвежда се по едно число на ред.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 100  99  80  70  Stop | 100 | -10  20  -30  Stop | 20 | 45  -20  7  99  Stop | 99 | 999  Stop | 999 | -1  -2  Stop | -1 |

## Най-малко число

Напишете програма, която до получаване на командата "**Stop**", чете **цели числа**, въведени от потребителя и намира **най-малкото** измежду тях. Въвежда се по едно число на ред.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 100  99  80  70  Stop | 70 | -10  20  -30  Stop | -30 | 45  -20  7  99  Stop | -20 | 999  Stop | 999 | -1  -2  Stop | -2 |