# Упражнения: Реализиране на цикличен алгоритъм

Състояние на настоящия учебен материал:



## Въпроси:

1. Какво представлява **цикълът**?
2. Дайте **примери** за **цикличност** от **ежедневието ни**.
3. Какви **циклични структури** познавате?
4. Какви са **разликите** между тези **циклични конструкции**?
5. Как може да **прекратим** **цикъл** в **Python**?
6. Какво прави операторът **continue**?

## Задачи:

### Числа от 1 до 100

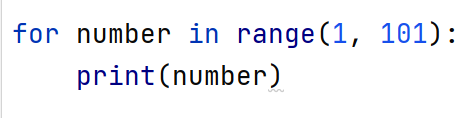
Напишете програма, която отпечатва числата от 1 до 100, всяко на нов ред.

#### Примерен вход и изход

|  |  |
| --- | --- |
| **вход** | **изход** |
| (няма) | 1  2  3  …  98  99  100 |

#### Насоки

Напишете решението на задачата, като си помогнете с кода от картинката по-долу:



### Числата от N до 1 в обратен ред

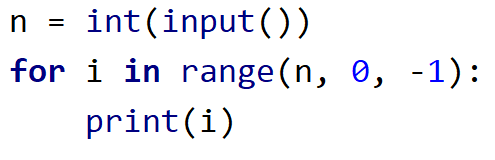
Напишете програма, която чете цяло положително число n, въведено от потребителя и печата **числата от** n **до** **1** **в обратен ред**. Въведеното число **n**, винаги ще бъде по-голямо от **1**.

#### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 2 | 2  1 | 3 | 3  2  1 | 5 | 5  4  3  2  1 |

#### Насоки

1. Прочетете **едно цяло число** от конзолата – броят на числата, които предстои да бъдат въведени
2. Направете for **цикъл от n до 0**, като **намалявате променливата i** **с 1** при **всяка итерация** и я принтирате.



### Поток от символи

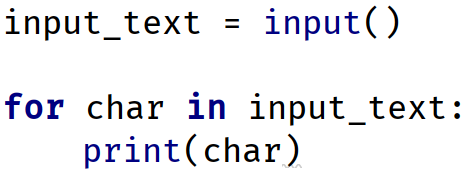
Напишете програма, която чете **текст** (стринг), въведен от потребителя и печата всеки **символ** от текста на отделен ред.

#### Примерен вход и изход

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

#### Насоки

За да итерира **for** цикълът през въведената дума на мястото на фунцията range(), използвайте директно променливата, в която преди това сте записали входа от конзолата:



### Сумиране на гласните букви

Да се напише програма, която чете **текст** (стринг), въведен от потребителя, и изчислява и отпечатва **сумата от стойностите на гласните букви** според таблицата по-долу:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| буква | a | e | i | o | u |
| стойност | 1 | 2 | 3 | 4 | 5 |

#### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **вход** | **изход** | **коментар** |
| 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 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.

#### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 |

### Редица цели числа

Напишете програма, която чете **n на брой цели числа**. Принтирайте **най-голямото** и **най-малкото** число сред въведените.

#### Примерен вход и изход

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |
| 5  10  20  304  0  50 | Max number: 304  Min number: 0 | 6  250  5  2  0  100  1000 | Max number: 1000  Min number: 0 |

### 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! |

#### Насоки

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



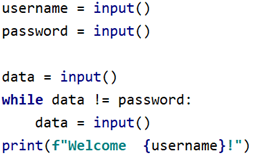
1. Инициализирайте променлива data, която ще държи въведената от потребителя парола за вход:



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



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



### 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

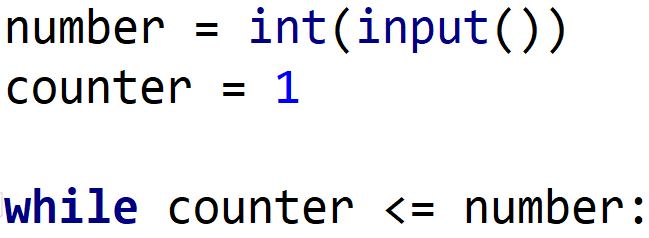
Напишете програма, която чете число n, въведено от потребителя,и отпечатва **всички числа ≤ n от редицата**: 1, 3, 7, 15, 31, …. Всяко следващо число се изчислява като умножим **предишното** с **2** и добавим **1**.

#### Примерен вход и изход

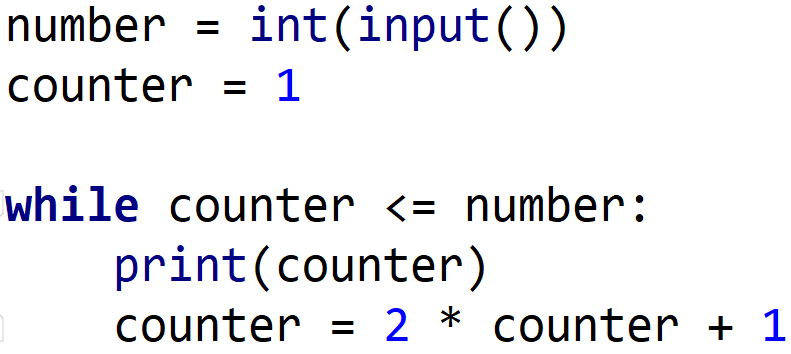
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 3 | 1  3 | 8 | 1  3  7 | 17 | 1  3  7  15 | 31 | 1  3  7  15  31 |

#### Насоки

1. Прочетете от конзолата цяло число;
2. **Създайте** променлива с **първоначална стойност** 1, която ще използвате за брояч:
3. **Създайте while** цикъл, който се повтаря, докато **броячът е по-малък** от числото, прочетено от конзолата:



1. При всяко повторение на цикъла **принтирайте стойността на брояча** и му **прибавяйте дадената стойност**:



### Най-малко число

Напишете програма, която до получаване на **командата "Stop",** чете **цели числа**, въведени от потребителя, намира **най-малкото** измежду тях и го принтира. 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 |

### Завършване

Напишете програма, която изчислява **средната оценка** на ученик от цялото му обучение. На първия ред ще получите **името на ученика**, а на всеки следващ ред неговите годишни оценки. Ученикът преминава в следващия клас, **ако годишната му оценка е по-голяма или равна на 4.00**. **Ако ученикът бъде скъсан повече от един път, то той бива изключен и програмата приключва,** като се отпечатва **името на ученика и в кой клас бива изключен**.

При успешно завършване на **12-ти** клас да се отпечата :

"{**име на ученика**} graduated. Average grade: {**средната оценка от цялото обучение**}"

**В случай, че ученикът е изключен от училище, да се отпечата:**

"{**име на ученика**} has been excluded at {**класа, в който е бил изключен**} grade"

**Стойността трябва да бъде форматирана до втория знак** след десетичната запетая.

#### Примерен вход и изход

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |
| Gosho  5  5.5  6  5.43  5.5  6  5.55  5  6  6  5.43  5 | Gosho graduated. Average grade: 5.53 | Mimi  5  6  5  6  5  6  6  2  3 | Mimi has been excluded at 8 grade |

### Монети

Производителите на вендинг машини искали да направят машините си да връщат възможно **най-малко монети ресто**. Напишете програма, която приема **сума** - **рестото**, което трябва да се върне и изчислява **с колко най-малко монети може да стане това**.

#### Примерен вход и изход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 1.23 | 4 | Рестото ни е 1 лев и 23 стотинки. Машината ни го връща с 4 монети: монета от 1 лев, монета от 20 стотинки, монета от 2 стотинки и монета от 1 стотинка. |
| 2 | 1 | Рестото ни е 2 лева. Машината ни го връща с 1 монета от 2 лева. |
| 0.56 | 3 | Рестото ни е 56 стотинки. Машината ни го връща с 3 монети: монета от 50 стотинки, монета от 5 стотинки и монета от 1 стотинка. |
| 2.73 | 5 | Рестото ни е 2 лева и 73 стотинки. Машината ни го връща с 5 монети: монета от 2 лева, монета от 50 стотинки, монета от 20 стотинки, монета от 2 стотинки и монета от 1 стотинка. |