# Упражнение 1: Работа със симвони низове и текст

## Преобразуване от 10-ична в N-ична ПБС

Напишете програма, която получава число в в 10-ична бройна система и го преобразува в число в N-ична бройна система, където 2 < = N < = 10. Входът се състои от 1 ред, съдържащ две числа, разделени с един интервал. Първото число е основа N, към която трябва да преобразувате. Вторият е число в 10-ична бройна система. Не използвайте никакви вградени функционалности за преобразуване на числа, опитайте се да напишете свой собствен алгоритъм.

**Упътване**

За алгоритъм (от 10-ична в 2-ична) можете да прочетете тази [статия](https://interactivepython.org/runestone/static/pythonds/BasicDS/ConvertingDecimalNumberstoBinaryNumbers.html).

Алгоритъмът за преобразуване на число от 10-ична в 2-ична бройна система е подобен: вместо “**% 2**”, ползвайте “**% N**”.

**Вход**

На един ред въвеждате основа на бройната система и число в 10-ична бройна система

**Изход**

На един ред извеждате числото в N-ична бройна система

**Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| 7 10 | 13 |
| 3 154 | 12201 |
| 5 123 | 443 |
| 4 1000 | 33220 |
| 9 3487 | 4704 |

## Преобразуване от N-ична в 10-ична ПБС

Напишете програма, която взема N-ично число и го преобразува 10-ично число (0 до 1050), където от 2 < = N < = 10. Входът се състои от 1 ред, съдържащ две числа, разделени с един интервал. Първото число е основата N, към която трябва да преобразувате. Второто е числото N, което трябва да се преобразува. Не използвайте никакви вградена функционалности за преобразуване, опитайте се да напишете свой собствен алгоритъм

**Вход**

На един ред въвеждате основа на бройната система и число в N-ична бройна система

**Изход**

На един ред извеждате числото в 10-ична бройна система

**Упътване**

Вижте тази картина за повече яснота за преобразуване от 2-ична в 10-ична БС. Отново, алгоритъмът за преобразуване от N-ична БС е подобен.

**Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| 7 13 | 10 |
| 3 12201 | 154 |
| 5 443 | 123 |
| 4 33220 | 1000 |
| 9 4704 | 3487 |

## Обръщане на низ

Създайте метод, който получава низ и връща низ, получен от същите символи, но в обратен ред.

**Вход**

На един ред поучавате символен низ

**Изход**

На един ред извеждате обърнатия низ

**Ограничения**

Символният низ да се състои от една дума, т.е. да няма интервали и да не се ползва метода Reverse

**Упътване**

Може да отпечатате всички символи на низа, като го обходите отзад напред или да конструирате нов низ, в който да прехвърлите символите на първия, в обратен ред

**Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| a | a |
| aba | aba |
| alenafanela | alenafanela |
| alibaba | ababila |
| baba | abab |

## Unicode Символи

Напишете програма, която преобразува символен низ в последователност от Unicode символни кодове.

**Вход**

На един ред въвеждате символен низ

**Изход**

На един ред извеждате Unicode на всеки символ

**Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| Hi! | \u0048\u0069\u0021 |
| What?!? | \0057\0068\0061\0074\003f\0021\003f |

## Умножаване на символни кодове

Създайте метод, който получава два низа като аргументи и връща сбора от техните произведения от символни кодове на съответни позиции (умножете str1.charAt (0) с str2.charAt (0) и ги добавете към сбора). След това продължете със следващите два знака. Ако един от низовете е по-дълъг от другия, добавете останалите символни кодове към сбора без умножение.

**Вход**

На един ред въвеждате два низа

**Изход**

На един ред извеждате сбора от техните произведения от символни кодове на съответни позиции

**Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| Gosho Pesho | 53253 |
| 123 522 | 7647 |
| a aaaa | 9700 |

## Палиндром

Създайте метод, който получава низ и връща True или False в зависимост от това дали думата е палиндром или не

**Вход**

На един ред поучавате символен низ

**Изход**

На един ред извеждате True, ако низа е палиндром или False, ако не е.

**Ограничения**

Символният низ да се състои от една дума, т.е. да няма интервали

**Упътване**

Един низ е палиндром, ако прочетен отзад напред е същия, какъвто е и когато го четем отпред назад

**Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| a | True |
| aba | True |
| alenafanela | True |
| alibaba | False |
| baba | False |