# More Exercises: Strings and Text Processing

Problems for exercises and homework for the ["Programming Fundamentals" course @ SoftUni](https://softuni.bg/trainings/3951/programming-fundamentals-with-java-january-2023).

You can check your solutions in [Judge](https://judge.softuni.org/Contests/1674/Text-Processing-More-Exercise).

## Extract Person Information

Write a program that reads **n** lines of strings and extracts the **name** and **age** of a given person. The person's name will be **between** "**@**"and"**|**". The person's **age** will be **between** "**#**"and"**\***".

**Example:** "**Hello my name is @Peter| and I am #20\* years old.**"

**For each** found name and age, **print** a line in the following format **"{name} is {age} years old."**

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  Here is a name @George| and an age #18\*  Another name @Billy| #35\* is his age | George is 18 years old.  Billy is 35 years old. |
| 3  random name @lilly| random digits #5\* age  @Marry| with age #19\*  here Comes @Garry| he is #48\* years old | lilly is 5 years old.  Marry is 19 years old.  Garry is 48 years old. |

## Ascii Sumator

Write a program that prints a **sum of all characters between two given characters** (their **ASCII code**). In the **first line,** you will get a **character**. In the **second line,** you get **another character**. On the **last line,** you get a **random string**. Find all the characters **between the two given** and **print their ASCII sum**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| .  @  dsg12gr5653feee5 | 363 |
| ?  E  @ABCEF | 262 |

## \*\*Treasure Finder

Write a program that **decrypts a message** by a given **key** and gathers information about the hidden **treasure type** and its **coordinates.** On the **first line,** you will receive a **key (sequence of numbers).** On the **next few lines, until you receive "find",** you will get lines of **strings**. You have to **loop through every string** and **decrease the ASCII code of each character** with a **corresponding number of the key** sequence. The way you choose a key number from the sequence is by just **looping through it**. If the **length of the key** sequence is **less than the string** sequence, you start **looping from the beginning of the key.** For more clarification, see the example below. **After decrypting** the message, you will **get a type of treasure and its coordinates.** The **type** will be **between** the symbol "**&**" and the coordinates will be between the symbols "**<**" and"**>**". For each line **print the type and the coordinates** in format **"Found {type} at {coordinates}".**

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comment** |
| 1 2 1 3  ikegfp'jpne)bv=41P83X@  ujfufKt)Tkmyft'duEprsfjqbvfv=53V55XA  find | Found gold at 10N70W  Found Silver at 32S43W | We start looping through the first string and the key. When we reach the end of the key, we start looping from the beginning of the key, but we continue looping through the string. (until the string is over)  The first message is: **"hidden&gold&at<10N70W>"** so we print we found gold at the given coordinates  We do the same for the second string  **"thereIs&Silver&atCoordinates<32S43W>"**(starting from the beginning of the key and the beginning of the string) |
| 1 2 2  nqtf(ujnxft(gtqn>Uphkb@  find | Found silver at Sofia |  |

## \*Morse Code Translator

Write a program that translates messages from **Morse code to English** (**capital letters).** Use [this](https://morsecode.world/international/morse2.html) page to help you (**without the numbers**). The words will be separated by a **space (' ')**. There will be a "**|**" character which you should **replace with ' '** (space).

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| .. | -- .- -.. . | -.-- --- ..- | .-- .-. .. - . | .- | .-.. --- -. --. | -.-. --- -.. . | I MADE YOU WRITE A LONG CODE |
| .. | .... --- .--. . | -.-- --- ..- | .- .-. . | -. --- - | -- .- -.. | I HOPE YOU ARE NOT MAD |