

My Ngram

Subject

1 Solution

Additional Resources
(1)

My Ngram

Remember to git add && git commit && git push each exercise!

We will execute your function with our test(s), please DO NOT PROVIDE ANY TEST(S) in your file

For each exercise, you will have to create a folder and in this folder, you will have additional files that contain your work. Folder names are provided at the beginning of each exercise under `submit directory` and specific file names for each exercise are also provided at the beginning of each exercise under `submit file(s)`.

My Ngram	
Submit directory	ex00
Submit file	my_ngram*
	It needs to be completed in the language you are working on right

Control Center



Group formation



In Progress



Submitted



Test review



Finished: approved



[Go To DoCode](#)



Access:

READ

WRITE



[Go To Gitea](#)



[Keep Working On This Solution](#)

**Also working on
the project**


[Learninghat](#) [Gitea](#) [Name - Login](#)


640

mirsalik_i



QPoints



Rust, ...

Description

SPECIFICATIONS

Write a program my_ngram; It will count the number of occurrences per character.

NAME

my_ngram

SYNOPSIS

my_ngram text [text2, text3]

DESCRIPTION

In computational linguistics and probability, an n-gram is a contiguous sequence of n items from a given sample of text or speech. The items can be phonemes, syllables, letters, words or base pairs according to the application. The n-grams typically are collected from a text or speech corpus. When the items are words, n-grams may also be called shingles.

Google Inc. has used this technique to improve the completion of its Search Engine. The program was developed by Jon Orwant and Will Brockman and released in mid-December 2010.

My Ngram will take 1 or multiple strings as arguments.

It will display, one per line, each character and the numbers of times it appears.

Order will be alphanumerical.

Example 00

[aubin_t](#)
[azimba](#)
[mirhoji](#)
[muxtor](#)

[ye_g](#)
[y_m](#)
[ov_j](#)

[rabbim](#)
[karimjo](#)
[risqibo](#)
[pardab](#)

[ov_j](#)
[n_d](#)
[y_i](#)
[oy_m](#)

[popan](#)
[tajiboy](#)
[ortiqbo](#)
[gaffaro](#)

[dop_a](#)
[e_a](#)
[y_a](#)
[v_s](#)

[panjiye](#)
[ismailo](#)
[temiro](#)
[ergash](#)

[v_s](#)
[v_u](#)
[v_a](#)
[e_mu](#)

Just finished

[otabek](#)
[bababe](#)
[frolova](#)
[nemat](#)

[ov_o](#)
[ko_j](#)
[_a](#)
[ov_a](#)

[hayrull](#)
[atabek](#)
[kamoli](#)
[abdulla](#)

[a_b](#)
[ov_a](#)
[dd_s](#)
[_be](#)

[aminjo](#)
[xolisov](#)
[umarb](#)
[abdazo](#)

[no_q](#)
[a_d](#)
[aye_x](#)
[v_m](#)

[olimjon](#)
[stupak](#)
[sodikov](#)
[tanasi](#)

[o_j](#)
[ov_k](#)
[_ar](#)
[n_t](#)

```
$>./my_ngram "abcdef"
a:1
b:1
c:1
d:1
e:1
f:1
$>
```

Example 01

```
$>./my_ngram "      "
:8
$>
```

8 spaces :-)

Example 02

```
$>./my_ngram "aaabb" "abc"
a:4
b:3
c:1
$>
```

Technical information:

1. (If you are doing this as project) you must create a Makefile, and the output is the command itself
2. You can use:


```
printf(3)
write(2)
```

3. You can NOT use:

Any functions/syscalls which does not appear in the previous list
Yes, it includes **exit**

Requirements

Your code must be compiled with the flags **-Wall -Wextra -**

 [ngo_t](#) [hwang_](#) [kanym](#) [nurlan](#)
[d](#) [kul_d](#) [ul_a](#)

Type
Project

Group
Size

2
Participants

Review
system

Test Review (Gandalf)

Difficult
y

Initiation

Averag
e
duratio
n

1
Week

Project's Metadata

Project

id: 347

name: my_ngram

visible: True

Werror.

Warnings

It's a bad practice to submit "object/binary files". Gandalf will reject your project if you submit your binary. (with the following message: "pushed file wrong format")

Ganfalf issue

Gandalf is sending an extra "", please add a if != from "" in order to pass the project.