



EXERCISES — Null Terminated Arrays

version #7be580532266ed398481e31366afcc24b1950c2a



**The way is lit. The path is clear.
We require only the strength to follow it.**

Copyright

This document is for internal use at EPITA ([website](#)) only.

Copyright © 2022-2023 Assistants [<assistants@tickets.assistants.epita.fr>](mailto:assistants@tickets.assistants.epita.fr)

The use of this document must abide by the following rules:

- ▷ You downloaded it from the assistants' intranet.*
- ▷ This document is strictly personal and must **not** be passed onto someone else.
- ▷ Non-compliance with these rules can lead to severe sanctions.

Contents

1	Goal	3
2	Example	3
2.1	Input:	3
2.2	Output:	4
2.3	Remarks	4

*<https://intra.assistants.epita.fr>

File Tree

```
null_terminated_arrays/  
├─ null_terminated_arrays.c  (to submit)  
└─ null_terminated_arrays.h  (to submit)
```

Authorized headers : You are only allowed to use the functions defined in the following headers

- err.h
- errno.h
- assert.h
- stddef.h

Compilation : Your code must compile with the following flags

- -std=c99 -pedantic -Werror -Wall -Wextra -Wvla

Main function : None

1 Goal

In this exercise you will manipulate arrays whose end is marked by a `NULL` pointer.

Write the function `reverse_matrix`, which takes an array of string arrays and reverses the main array as well as all the nested arrays, you should not reverse the string itself.

```
void reverse_matrix(const char ***matrix);
```

`matrix` is a null-terminated array of null-terminated string arrays. The strings stored in each array are valid and null-terminated strings.

- `matrix (const char **)` is ended by a `NULL` pointer : the array of arrays.
- `matrix[0] (const char *)` is ended by a `NULL` pointer : the array of strings.
- `matrix[0][0] (const char *)` is ended by a `\0` character: the string (array of chars).

This means that in order to find the size of an array, you have to find the `NULL` element that marks the end of the array.

2 Example

2.1 Input:

```
{  
  { "1", "2", "3", NULL },  
  { "4", "5", NULL },  
  { "6", "7", "8", "9", NULL },  
}
```

(continues on next page)

```
    NULL  
}
```

2.2 Output:

```
{  
  { "9", "8", "7", "6", NULL },  
  { "5", "4", NULL },  
  { "3", "2", "1", NULL },  
  NULL  
}
```

2.3 Remarks

- Notice how the NULL still marks the end of the array.
- None of the strings can be NULL, because that would be breaking the end marker.
- Do not reverse the string itself.

The way is lit. The path is clear. We require only the strength to follow it.