Exercises — King of the Hill, array ec

version **#**

Assistants C/Unix 2022 <assistants@tickets.assistants.epita.fr>

Copyright

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

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

1 King of the Hill, array edition

Files to submit:

- hill_array/hill_array.c

Authorized functions: You are only allowed to use the following functions:

- printf(3)

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

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

1.1 Goal

We consider that an array is a *hill* if is is composed of:

- A sequence of positive increasing integers (`<n+1>` being superior or equal to `<n>`)
- The *Top of the hill*: one or several equal integers.
- A sequence of positive decreasing integers (`<n+1>` being inferior or equal to `<n>`).

For example, this is a valid *hill*:

```
int arr[] =
{
    1, 2, 3, 4, 5, 6, 7, 7, 2, 1, 0, 0
};
```

And this is **not** a hill:

```
int arr[] =
{
    0, 2, 3, 2, 4, 5, 4, 3, 7, 1
};
```

Write a function that takes as input an array of `int` and its length, and returns the index of the *top of the hill*. You must also check that the *hill* is correct.

If the hill is invalid, the function returns `-1`. An empty array is considered invalid.

```

```

With an array like this:

```
int arr[] =
{
    0, 2, 3, 4, 6, 7, 9, 9, 7, 6, 5, 4, 2, 1
};
```

The top of the hill is the first 9, thus you must return the index 6.

1.2 Prototype

```
int top_of_the_hill(int tab[], size_t len)
```

1.3 Examples

```
int main(void)
{
    int tab1[] = { 1, 2, 3, 4, 6, 6, 4, 2, 0, 0 }; // Valid hill.

    printf("%d\n", top_of_the_hill(tab1, 10));

    int tab2[] = { 1, 2, 3, 4, 5, 6, 6, 6, 6, 6 }; // Valid hill.

    printf("%d\n", top_of_the_hill(tab2, 10));

    int tab3[] = { 1, 2, 3, 4, 6, 6, 4, 5, 0, 0 }; // Invalid hill.

    printf("%d\n", top_of_the_hill(tab3, 10));

    return 0;
}
```

```
42sh$ ./hill_array | cat -e
4$
5$
-1$
```

It is my job to make sure you do yours.