EXERCISES — String replacement

version #

ASSISTANTS C/UNIX 2022 <assistants@tickets.assistants.epita.fr>

Copyright

Contents

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

1 String replacement

Files to submit:

- string_replace/string_replace.c
- string_replace/string_replace.h

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

- malloc(3)
- calloc(3)
- free(3)

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

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

1.1 Goal

You have to implement a function that copies the string `str` to a newly allocated string while replacing each `c` character with the string `pattern`. The function returns the new string.

```
char *string_replace(char c, const char *str, const char *pattern);
```

- The `c` character cannot be `\0`.
- If `pattern` is `NULL`, every occurrence of the `c` character will be removed in the new string.
- If `str` is `NULL`, `string_replace` returns `NULL`.

1.2 Example

```
#include <stdio.h>
#include <stdlib.h>

#include "string_replace.h"

void check(char c, const char *s, const char *p)
{
    char *res = string_replace(c, s, p);

    printf("%s\n", res);
    free(res);
}

```

(continues on next page)

(continued from previous page)

```c
int main(void)
{
    check('o', "bobo", "");
    check('o', "bobo", "i");
    check('o', "bobo", "oo");

    return 0;
}
```

```
42sh$ gcc -Wall -Wextra -Werror -std=c99 -pedantic -o replace string_replace.c main.c
42sh$ ./replace | cat -e
bb$
bibi$
booboo$
```

It is my job to make sure you do yours.