

## Original code

```
#include "stdlib.h"
#include "string.h"
#include "assert.h"
char* stringDuplicator(char* s, int times){
    assert(!s);
    assert(times > 0);
    int LEN = strlen(s);
    char* out = malloc(LEN*times);
    assert(out);
    for (int i=0; i<=times; i++){
        out = out + LEN;
        strcpy(out,s);
    }
    return out;
}
```

## List of mistakes

### Code conventions:

1. Capital letters shouldn't be used for "LEN" parameter.
2. Inside the for loop, the lines had no indentation.

### Coding mistakes:

1. For condition should be  $i < \text{times}$ , not  $\leq$ .
2. Include syntax should have `<>` others then `""`.
3. The order of lines inside the for loop should be reordered.
4. In the original code, "out" pointer is changed inside the for loop and at the end when returned, will point at the end of the string instead of the begging. Therefore, added "temp" pointer which would be changed only.
5. In the instructions, it is specified that we should return NULL in case of any failure. Any assert in the original code would just break without returning NULL, therefore any assert replaced with "if" and returns NULL.
6. In the original code, there isn't `'\0'` at the end of the string.

## Edited code

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
char *stringDuplicator(char *s, int times) {
    if (s == NULL || times <= 0) {
        return NULL;
    }
    int len = strlen(s);
    char *out = malloc(len * times + 1); //+1 for the \0 char.
    if (out == NULL) {
        return NULL;
    }
    char *temp = out;
    for (int i = 0; i < times; i++) {
        strcpy(temp, s);
        temp = temp + len;
    }
    *(temp+1) = '\0';
    return out;
}
```