CSL 301 - Operating System
XV6 - Lab 1

Your turn to add a system call- "Syscall to Reverse a String from the User `strrev()`"

What it does - Takes a string buffer from user-space, reverses it in place.

Step 1: Update `syscall.h`

Assign a unique number (e.g., 25):

```
#WRITE YOUR CODE HERE
```

Step 2: Update `syscall.c`

Declare your handler and add to the syscall table:

```
#WRITE YOUR CODE HERE
```

Add to the syscall table:

```
#WRITE YOUR CODE HERE
```

Step 3: Implement in `sysproc.c`

```
int sys_strrev(void) {
        char *str;
        int len, i;
        if (argstr(0, &str) < 0 || argint(1, &len) < 0)
        return -1;

        // Simple in-place reversal
        for (i = 0; i < len/2; i++) {
                char tmp = str[i];
                str[i] = str[len - i - 1];
                str[len - i - 1] = tmp;
        }
        return 0;
}
```

Step 4: Add Prototype to `user.h` named as `strrev()`

return type for `strrev()` is int and it takes two parameters - 1. pointer to char and 2. integer parameter for length

```
#WRITE YOUR CODE HERE
```

Step 5: Add Assembly Stub to `usys.S`

```
#WRITE YOUR CODE HERE
```

Step 6: User Test Program (`teststrrev.c`)

```
#include "types.h"
#include "stat.h"
#include "user.h"

int main(void) {
        char buf[32] = ‘‘Write your full name here’’;  //this
string will be passed to our newly created system call
        printf(1,‘‘Before: %s\n’’, buf);
        strrev(buf, strlen(buf)); //passing string and string
length
        printf(1, ‘‘After: %s\n’’, buf);
        exit();
}
```

Add to your Makefile's `UPROGS` as `_teststrrev`.