

# mypwd

man 2 getcwd read carefully

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

int main() {
    char *buf=getcwd(NULL, 0);    //here getcwd dynamiclly alloc the buf for
    varying size
    if (buf==NULL){
        printf("process failed\n");
        exit(-1);
    }
    size_t len = strlen(buf);    //getting size for writing
    if (write(1, buf,len) < 0) {
        printf("Write failed\n");
        exit(-3);
    }
    write(1, "\n", 1);
    free(buf);    // re free the alloc
    return 0;
}
```

# myecho

carefully deal with args

```
#include <unistd.h>    // for write(), read()
#include <stdlib.h>    // for exit()

// Manual string length function (no strlen)
size_t my_strlen(const char *s) {
    size_t len = 0;
    while (s[len] != '\0') len++;
    return len;
}

int main(int argc, char *argv[]) {
    if (argc <= 1) {
```

```

        // No arguments -> just newline
        write(1, "\n", 1);
        return 0; // success
    }

    for (int i = 1; i < argc; i++) {
        size_t len = my_strlen(argv[i]);
        if (len > 0) {
            write(1, argv[i], len);
        }
        if (i < argc - 1) {
            // Add space between arguments
            write(1, " ", 1);
        }
    }
    // Always end with newline
    write(1, "\n", 1);

    return 0; // success always
}

```

## mycp

**create new file with permissions**

```

#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#define COUNT 100
int main(int argc, char *argv[])
{
    char buf[COUNT];

    if (argc != 3) {
        printf("Usage: %s file-name1 file-name2\n", argv[0]);
        exit(-4);
    }

    int fdrd = open(argv[1], O_RDONLY);
    if (fdrd < 0) {
        printf("could not open a to read\n");
        exit(-1);
    }
}

```

```

    int fdwr = creat(argv[2], S_IRUSR | S_IWUSR | S_IRGRP | S_IROTH);
    if (fdwr < 0) {
        printf("could not open a write\n");
        exit(-2);
    }

    int num_read;
    while ((num_read = read(fdrd, buf, COUNT)) > 0) {
        if (write(fdwr, buf, num_read) < 0) {
            printf("Write failed\n");
            exit(-3);
        }
    }

    close(fdrd);
    close(fdwr);

    return 0;
}

```

## mymv

### man 2 unlink

```

#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#define COUNT 100
int main(int argc, char *argv[])
{
    char buf[COUNT];

    if (argc != 3) {
        printf("Usage: %s file-name1 file-name2\n", argv[0]);
        exit(-4);
    }

    int fdrd = open(argv[1], O_RDONLY);
    if (fdrd < 0) {
        printf("could not open a to read\n");
        exit(-1);
    }
}

```

```
int fdwr = creat(argv[2], S_IRUSR | S_IWUSR | S_IRGRP | S_IROTH);
if (fdwr < 0) {
    printf("could not open a write\n");
    exit(-2);
}
```

```
int num_read;
while ((num_read = read(fdrd, buf, COUNT)) > 0) {
    if (write(fdwr, buf, num_read) < 0) {
        printf("Write failed\n");
        exit(-3);
    }
}
```

```
close(fdrd);
close(fdwr);
unlink(argv[1]);
return 0;
```

```
}
```