Examples

- Read 80 bytes from a file.

```
enum {MAX_LEN = 80};
int num;
FILE *fptr2;
char filename2[]= "haiku.txt";
char buff[MAX_LEN + 1];
if ((fptr2 = fopen(filename2, "r")) == NULL){
  printf("Cannot open %s.\n", filename2);
  reval = FAIL; exit(1);
}
. . . .
num = fread(buff, sizeof(char), MAX_LEN, fin);
buff[num * sizeof(char)] = `\0';
printf("%s", buff);
```

Exercise

- Write a program that use bloc-based file operations to copy the content of lab1.txt to to lab1a.txt
- Use: fread, fwrite, feof

Solution

```
#include <stdio.h>
enum {SUCCESS, FAIL, MAX_LEN = 80};
void BlockReadWrite(FILE *fin, FILE *fout);

main(void) {
        FILE *fptr1, *fptr2;
        char filename1[]= "lab1a.txt";
        char filename2[]= "lab1.txt";
        int reval = SUCCESS;

        if ((fptr1 = fopen(filename1, "w")) == NULL){
                printf("Cannot open %s.\n", filename1);
                reval = FAIL;
        } else if ((fptr2 = fopen(filename2, "r")) == NULL){
        printf("Cannot open %s.\n", filename2);
        reval = FAIL;
        } else {
         BlocReadWrite(fptr2, fptr1);
         fclose(fptr1);
         fclose(fptr2);
        }
        return reval;
}
```

Solution

```
void BlockReadWrite(FILE *fin, FILE *fout) {
    int num;
    char buff[MAX_LEN + 1];

    while (!feof(fin)){
     num = fread(buff, sizeof(char),
         MAX_LEN, fin);
     buff[num * sizeof(char)] = `\0';

     printf("%s", buff);
     fwrite(buff, sizeof(char), num, fout);
    }
}
```

Exercise

- A)Improve the program in previous exercise so that it accepts the two filenames as command arguments.

- For example: if your program is named "filecpy". You can use it as the following syntax (in Linux):
- ./filecpy haiku.txt haiku2.txt

- B. Write a program having the same functionality as cat command in Linux
- ./cat1 haiku.txt

Hint

- Just use the argc[] et argv[]

```
if(argc<3) { printf("%s <file1> <file2>n",argv[0]); exit(1); }
```

- argv[1] and argv[2] will be the name of source file and destination file.

```
if((fp=fopen(argv[1],"r"))==NULL) {
…
};
if((fp2=fopen(argv[2],"w"))==NULL) {
…
};
```

Solution

```
#include <stdio.h>
enum {SUCCESS, FAIL, MAX_LEN = 80};
void BlockCat(FILE *fin);

main(int argc, char* argv[]) {
        FILE *fptr1, *fptr2;
        int reval = SUCCESS;
        if (argc !=2){
                printf("The correct syntax should be: cat1
                        filename \n");
                reval = FAIL;
        }

        if ((fptr1 = fopen(argv[1], "r")) == NULL){
                printf("Cannot open %s.\n", argv[1]);
                reval = FAIL;
        } else {
        BlocCat(fptr1);
        fclose(fptr1);
        }
        return reval;
}
```

Solution

```
void BlockCat(FILE *fin) {
     int num;
     char buff[MAX_LEN + 1];

     while (!feof(fin)){
      num = fread(buff, sizeof(char),
          MAX_LEN, fin);
      buff[num * sizeof(char)] = `\0';

      printf("%s", buff);
     }
}
```