Introduction to Scientific and Engineering Computation (BIL 104E)

Lab 13

Reading and Writing with Files: Closing a File

```
#include <stdio.h>
enum {SUCCESS, FAIL};
main(void)
{
   FILE *fptr;
   char filename[]= "haiku.txt";
   int reval = SUCCESS;

   if ((fptr = fopen(filename, "r")) == NULL){
     printf("Cannot open %s.\n", filename);
     reval = FAIL;
   }
   else {
       printf("The value of fptr: 0x%p\n", fptr);
       printf("Ready to close the file.");
       fclose(fptr);
   }
   getchar();

   return reval;
}
```

```
The value of fptr: 0x760609B8
Ready to close the file.
```

Reading and Writing Disk Files:One Character at a Time

```
#include <stdio.h>
enum {SUCCESS, FAIL};
void CharReadWrite(FILE *fin, FILE *fout);
main(void)
   FILE *fptr1, *fptr2;
   char filename1[]= "outhaiku.txt";
   char filename2[]= "haiku.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 {
           CharReadWrite(fptr2, fptr1);
           fclose(fptr1);
           fclose(fptr2);
       }
   getchar();
   return reval;
}
```

```
/* function definition */
void CharReadWrite(FILE *fin, FILE
*fout)
{
   int c;

   while ((c=fgetc(fin)) != EOF){
        fputc(c, fout); /* write to a file */
        putchar(c); /* put the character to the
screen */
   }
}
```

Leading me along
my shadow goes back home
from looking at the moon.
--- Sodo
(1641-1716)

A storm wind blows
out from among the grasses
the full moon grows.
--- Chora
(1729-1781)

Reading and Writing Disk Files:One Line at a Time

```
#include <stdio.h>
enum {SUCCESS, FAIL, MAX_LEN = 81};
void LineReadWrite(FILE *fin, FILE *fout);
main(void)
{   FILE *fptr1, *fptr2;
   char filename1[]= "outhaiku.txt";
   char filename2[]= "haiku.txt";
   int reval = SUCCESS;
   if ((fptr1 = fopen(filename1, "w")) == NULL){
     printf("Cannot open %s for writing.\n", filename1);
      reval = FAIL;
   }
   else
       if ((fptr2 = fopen(filename2, "r")) == NULL){
          printf("Cannot open %s for reading.\n", filename2);
          reval = FAIL;
       }
       else {
           LineReadWrite(fptr2, fptr1);
           fclose(fptr1);
           fclose(fptr2);
       }
    getchar();
    return reval;
}
```

```
/* function definition */
void   LineReadWrite(FILE   *fin,   FILE
*fout)
{
   char buff[MAX_LEN];

  while (fgets(buff, MAX_LEN, fin) != NULL){
        fputs(buff, fout);
        printf("%s", buff);
   }
}
```

Leading me along
my shadow goes back home
from looking at the moon.
--- Sodo
(1641-1716)

A storm wind blows
out from among the grasses
the full moon grows.
--- Chora
(1729-1781)

Reading and Writing Disk Files: feof() function

```
#include <stdio.h>
enum {SUCCESS, FAIL, MAX_LEN = 80};
void BlockReadWrite(FILE *fin, FILE *fout);
int ErrorMsg(char *str);
main(void)
{
   FILE *fptr1, *fptr2;
   char filename1[]= "outhaiku.txt";
   char filename2[]= "haiku.txt";
   int reval = SUCCESS;
   if ((fptr1 = fopen(filename1, "w")) == NULL){
     reval = ErrorMsg(filename1);
   }
   else
       if ((fptr2 = fopen(filename2, "r")) == NULL){
          reval = ErrorMsg(filename2);
       }
       else {
           BlockReadWrite(fptr2, fptr1);
           fclose(fptr1);
           fclose(fptr2);
       }
   getchar();
   return reval;
}
```

Reading and Writing Disk Files: feof() function

```
/* function definition */
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'; /* append a null character */
        printf("%s", buff);
        fwrite(buff, sizeof(char), num, fout);
   }
}
/* function definition */
int ErrorMsg(char *str)
{
   printf("Cannot open %s.\n", str);
   return FAIL;

}
```

Reading and Writing Disk Files: feof() function

Computer Screen

Leading me along
my shadow goes back home
from looking at the moon.
--- Sodo
(1641-1716)

A storm wind blows
out from among the grasses
the full moon grows.
--- Chora
(1729-1781)

Special File Functions: fscanf() and fprintf()

```
#include <stdio.h>
enum {SUCCESS, FAIL,
     MAX_NUM = 3,
     STR_LEN = 23};
void DataWrite(FILE *fout);
void DataRead(FILE *fin);
int ErrorMsg(char *str);
main(void)
{
FILE *fptr;
    char filename[]= "strnum.mix";
    int reval = SUCCESS;

    if ((fptr = fopen(filename, "w+")) == NULL){
       reval = ErrorMsg(filename);
    }
    else {
         DataWrite(fptr);
         rewind(fptr);
         DataRead(fptr);
         fclose(fptr);
         }
    getchar();
    return reval;
}
```

Special File Functions: fscanf() and fprintf()

```
/* function definition */
void DataWrite(FILE *fout)
{
   int i;
   char cities[MAX_NUM][STR_LEN] = {
                        "St.Louis->Houston:",
                        "Houston->Dallas:",
                        "Dallas->Philadelphia:"};
   int miles[MAX_NUM] = {
                    845,
                    243,
                    1459};

   printf("The data written:\n");
   for (i=0; i<MAX_NUM; i++){
      printf("%-23s %d miles\n", cities[i], miles[i]);
      fprintf(fout, "%s %d", cities[i], miles[i]);
   }
}
```

Special File Functions: fscanf() and fprintf()

```
/* function definition */
void DataRead(FILE *fin)
{
   int i;
   int miles;
   char cities[STR_LEN];

   printf("\nThe data read:\n");
   for (i=0; i<MAX_NUM; i++){
      fscanf(fin, "%s%d", cities, &miles);
      printf("%-23s %d miles\n", cities, miles);
   }
}
/* function definition */
int ErrorMsg(char *str)
{
   printf("Cannot open %s.\n", str);
   return FAIL;
}
```

Special File Functions: fscanf() and fprintf()

Computer Screen

```
The data written:
St.Louis->Houston:      845 miles
Houston->Dallas:        243 miles
Dallas->Philadelphia:   1459 miles

The data read:
St.Louis->Houston:      845 miles
Houston->Dallas:        243 miles
Dallas->Philadelphia:   1459 miles
```