(Standard) Input and Output

Input
Keyboard, file, network

Output
Screen, file, printer, network

Abstraction

```
int  scanf(        char *format, ...)
int fscanf(FILE *, char *format, ...)
int sscanf(char *, char *format, ...)

int  printf(        char *format, ...)
int fprintf(FILE *, char *format, ...)
int sprintf(char *, char *format, ...)
```

```
fp = fopen("TENLINES.TXT","w");
$ proj < myinputfile.txt

$ dir > myfilelist.txt

stdin, stdout, stderr

#include "stdio.h"
int
main( )
{
     FILE *funny;
     int c;
     funny = fopen("TENLINES.TXT","r");
     if (funny == NULL)
          printf("File doesn't exist\n");
     else
     {
          do
          {
               c = getc(funny); /* get one character from the file */
               putchar(c); /* display it on the monitor */
          } while (c != EOF); /* repeat until EOF (end of file) */
          fclose(funny);
     }
}
```

```
#include "stdio.h"
int
main( )
{
      FILE *fp1;
      char oneword[100];
      int c;
      fp1 = fopen("TENLINES.TXT","r");
      do
      {
            c = fscanf(fp1,"%s",oneword); /* got one word from the file */
            printf("%s\n",oneword); /* display it on the monitor */
      } while (c != EOF); /* repeat until EOF */
      fclose(fp1);
}

#include "stdio.h"
#define MAXLINE 1000
int
main( )
{
      FILE *fp1;
      char oneline[MAXLINE];
      char *c;
      fp1 = fopen("TENLINES.TXT","r");
      do
      {
            c = fgets(oneline, MAXLINE,fp1); /* get one line from the file */
            if (c != NULL);
                  printf("%s",oneline); /* display it on the monitor */
      } while (c != NULL); /* repeat until NULL */
      fclose(fp1);
}

#include <stdio.h>
#include <ctype.h>
main() /* lower: convert input to lower case*/
{
      int c
      while ((c = getchar()) != EOF)
            putchar(tolower(c));
      return 0;
}

void filecopy(FILE *ifp, FILE *ofp)
{
      int c;
      while ((c = getc(ifp)) != EOF)
            putc(c, ofp);
}
```

```
#include <stdio.h>
main()
{
     double sum, v;
     sum = 0;
     while (scanf("%lf", == 1)
          printf("\t%.2f\n", sum += v);
     return 0;
}

FILE *fp;
FILE *fopen(char *name, char *mode);
fp = fopen(name, mode)
read("r"), write ("w"), and append ("a").

if ((fp = fopen(myfilename, "r")) == NULL)
     ...

fprintf(stderr, "%s: error writing stdout\n", prog);

char *fgets(char *line, int maxline, FILE *fp)
int fputs(char *line, FILE *fp)


char *fgets(char *line, int maxline, FILE *fp)
fgets reads the next input line (including the newline) from file fp
     into the character array line; at most
maxline-1 characters will be read. The resulting line is terminated with
'\0'. Normally fgets returns line; on end of file or error it returns NULL.
(Our getline returns the line length,  which is a more useful value; zero
means end of file.)

For output, the function fputs writes a string (which need not contain a
newline)
to a file: int fputs(char *line, FILE *fp) It returns EOF if an error
occurs,
and non-negative otherwise.



/* getline: read a line, return length */
int getline(char *line, int max)
{
     if (fgets(line, max, stdin) == NULL)
          return 0;
     return strlen(line);
}
```

```
...
while  (getline(line, MAXLINE) != 0)
{
     printf("Line is |%s|\n",line);

     line[strlen(line)] = '\0';

     printf("Line is |%s|\n",line);


     sscanf(line,"%d",&mynum);
     sum = sum + mynum;

     sscanf(line, "%d %d",&mynum1, &mynum2);

}
```

```
#include "stdio.h"
#define MAXLINE 1000
int
main( )
{
     FILE *fp1;
     char oneline[MAXLINE];
     char *c;
     int nlines = 0;
     int sum = 0;
     fp1 = fopen("TENLINES.TXT","r");

     if (fp1 == NULL)
          printf("No file\n");
     else
     {
          do
          {
               c = fgets(oneline, MAXLINE,fp1); /* get one line from the
          file */
               if (c != NULL);
                    printf("%s",oneline); /* display it on the monitor */
               oneline[strlen(oneline)-1] = '\0';
               sum += atoi(oneline);
               nlines++;
          } while (c != NULL); /* repeat until NULL */
          fclose(fp1);
          printf("mean is %lf \n", (    (double) sum  ) / ( (double) n) );
     }
}
```

```
#include "stdio.h"
#define MAXLINE 1000
int
main( )
{
     FILE *fp1, *fp2;
     char oneline1[MAXLINE];
     char oneline2[MAXLINE];

     char *c1, *c2;
     int nlines = 0;
     fp1 = fopen("file1.TXT","r");
     fp2 = fopen("file2.TXT","r");


     if ((fp1 == NULL) || (fp2 == NULL))
          printf("No file\n");
     else
     {
          do
          {
               c1 = fgets(oneline1, MAXLINE,fp1); /* get one line from the
          file */
               c2 = fgets(oneline2, MAXLINE,fp2); /* get one line from the
          file */

               if ((c1 != NULL) && (c2 != NULL))
               {
                    if (strcmp(c1,c2) == 0)
                         nlines++;

               }
          } while ((c1 != NULL) && ((c2 != NULL))); /* repeat until NULL */
          fclose(fp1);
          fclose(fp2);

          printf("equal lines are %d\n",nlines);

     }
}
```