`\'`	Single quote character
`\"`	Double quote character
`\?`	Question mark character
`\\`	Backslash
`\a`	Audible bell (or visual alert)
`\b`	Backspace
`\f`	Form feed
`\n`	Newline
`\r`	Return to beginning of line
`\t`	Horizontal tab
`\v`	Vertical tab

Formatting input with scanf

- `scanf` uses a different set of conversion specifiers compared to `printf`

`%d`	Signed or unsigned decimal integer
`%i`	Signed or unsigned decimal, octal, or hexadecimal integer
`%o`	Unsigned octal integer
`%u`	Unsigned decimal integer
`x` or `X`	Unsigned hexadecimal integer
`%h` or `l`	Used with any of the above integer conversion specifiers to indicate a `long` or `short` integer
`e`, `E`, `f`, `g`, or `G`	Floating point number
`l` or `L`	double or `long double`
`%c`	Single character
`s`	A string, terminated by `NULL`
scan set	Scan the string of characters stored in the format
`%`	Skip a percent character in the input

%e or %E	Print floating point number in exponent notation
%f	Print floating point number in fixed point notation
%g or %G	Print floating point number in fixed point notation (%f) if the value does not fit, print it in exponent notation (%e)
%Ld	Print a `long double` floating point number

Printing strings and characters

- Use the following conversion specifiers

%c	Print a character
%s	Print a string (array of characters terminated by `NULL`

- A string must be terminated by a `NULL`
- You must enclose a character constant in single quotes and a string constant in double quotes

Printing with field widths and precisions

- An integer inserted between the percent character and the conversion specifier indicates the width of the field used to print the variable value
- All variables printed with a specified field width are right justified
- Precision values can also be specified in the argument list by replacing the field width specifiers with asterisks as follows

```
printf ( "%*.*f", 7, 2, 98.736 );
```

  The result is 98.74 right justified (with two leading spaces)

Using flags in the printf format control string

- Used to supplement the formatting control abilities of `printf`
- The flags are defined as follows

-	Print the output left-justified
+	Display a plus or minus sign as appropriate
space	Print a space before a positive value not printed with the + flag
#	Prefix `0` to an octal number Prefix `0x` or `0X` to a hexadecimal number Force a decimal point for a floating point whole number
0	Pad a field with leading zeros

- You may combine several flags in one conversion specification

Printing escape sequences

- Escape sequences are used to print special characters like newline

Formatted Input/Output

Streams

- Communication channels between files (devices) and programs
- No distinction between files and I/O devices
- Used to perform input and output
- The name stream comes from a "stream of bytes" – the way C looks at files and devices
- Three standard streams automatically get connected to the process
  - `stdin` – Standard input connected to the keyboard
  - `stdout` – Standard output connected to the monitor
  - `stderr` – Standard error connected to the monitor
- Any stream can be redirected to other files
- In C shell, the `stderr` can be redirected by using the following

```
( cmd > file1 ) >& file2
```

Formatting output with printf

- Every `printf` statement contains a format string to describe the output format
- A call to the function `printf` has the form

```
printf ( format_string, arg1, arg2, ... )
```

  where zero or more arguments may follow the first argument `format_string`
- The format string must be enclosed in double quotes

Printing integers

- Printed by using the conversion specifiers as per the following table

`%d`	Signed decimal integer
`%i`	Signed decimal integer
`%o`	Unsigned octal integer
`%u`	Unsigned decimal integer
`%x` or `%X`	Unsigned hexadecimal integer
`%hd`	Print `short` integer
`%ld`	Print `long` integer

- Do not print a negative value with a conversion specifier that expects an unsigned value

Printing floating-point numbers

- Can be printed in decimal point notation or exponent notation