5 Terminal Output

5.1 Formatting output with *printf*

printf (*format-control-string, other-arguments*)

format-control-string describes the output format, which consists of conversion specifiers, field widths, precisions and literal characters with percent sign(%).

Conversion specifier	Description
`d`	Display as a *signed decimal integer*.
`i`	Display as a *signed decimal integer*. [*Note:* The `i` and `d` specifiers are *different* when used with `scanf`.]
`o`	Display as an *unsigned octal integer*.
`u`	Display as an *unsigned decimal integer*.
`x` or `X`	Display as an *unsigned hexadecimal integer*. `X` causes the digits `0-9` and the *uppercase* letters `A-F` to be used in the display and `x` causes the digits `0-9` and the *lowercase* letters `a-f` to be used in the display.
`h`, `l` or `ll` (letter "ell")	Place *before* any integer conversion specifier to indicate that a `short`, `long` or `long long` integer is displayed, respectively. These are called length modifiers.
`e` or `E`	Display a floating-point value in *exponential notation*.
`f` or `F`	Display floating-point values in *fixed-point notation* (`F` is supported in the Microsoft Visual C++ compiler in Visual Studio 2015 and higher).
`g` or `G`	Display a floating-point value in either the *floating-point form* `f` or the exponential form `e` (or `E`), based on the magnitude of the value.
`L`	Place before any floating-point conversion specifier to indicate that a `long double` floating-point value should be displayed.

Type	Format Specifier
int	%d
char	%c
float	%f
double	%lf
short int	%hd
unsigned int	%u
long int	%li
long long int	%lli
unsigned long int	%lu
unsigned long long int	%llu
signed char	%c
unsigned char	%c
long double	%Lf

Example:

```
int a=1234;
float f=123.456;
char ch='a';
printf("%8d,%2d\n",a,a);
printf("%f,%8f,%8.1f,%.2f,%.2e\n",f,f,f,f,f);
printf("%3c\n",ch);
```

Sample output:

```
    1234,1234
123.456000,123.456000,    123.5,123.46,1.23e+02
   a
```

5.2 *cout*

cout << variable1(expression1) [<< variable2 << variable n];

CPP > coutdemo.cpp

```
4   int main()
5   {
6       int a = 10;
7       float b = 45.7;
8       char c = 'A';
9
10      cout << "a = " << a << ",b = " << b << ",c = " << c << endl;
11
12      return 0;
13  }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
liao@DESKTOP-OOC4F37:/mnt/e/CCode$ cd CPP
liao@DESKTOP-OOC4F37:/mnt/e/CCode/CPP$ g++ coutdemo.cpp
liao@DESKTOP-OOC4F37:/mnt/e/CCode/CPP$ ./a.out
a = 10,b = 45.7,c = A
liao@DESKTOP-OOC4F37:/mnt/e/CCode/CPP$
```