Arab Academy for Science and Technology and Maritime Transport

College of Engineering and Technology

Computer Engineering Department

CC112 Structured Programming

Lecture 3

LECTURE 3

Input / output operations

LECTURE OUTLINE

i. **Input/Output operations Function**
   - **Output function**
   - **Place holders**
   - **Escape sequences**
   - **Input function**
   - **Output formatting**

ii. **Programming Examples**

I. INPUT/OUTPUT OPERATIONS FUNCTION

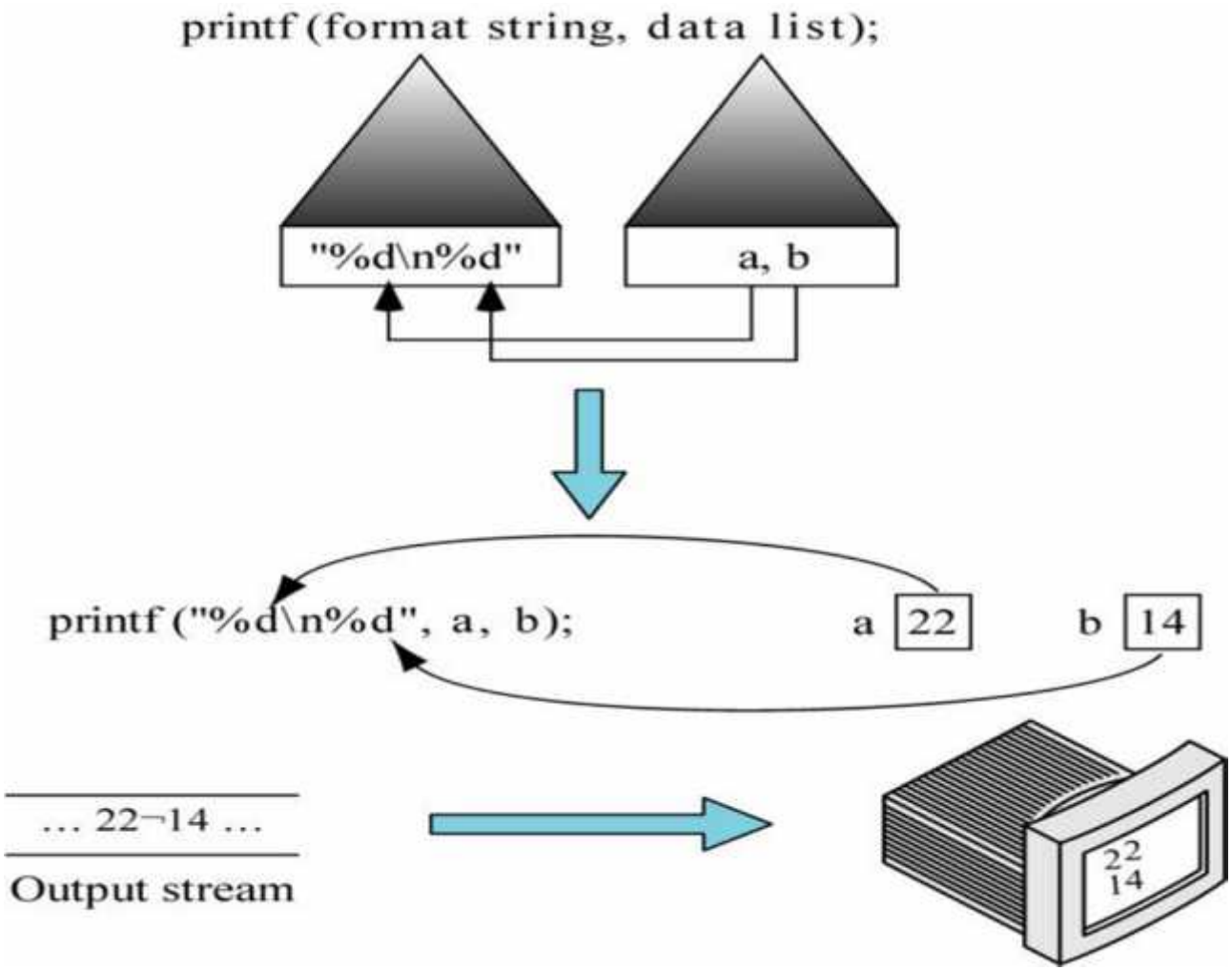
Output Function (printf)

The *printf* function displays the value of its format string and the values of the expressions in the print list in the same order from left-to-right. They are given between two double quotations.

SYNTAX

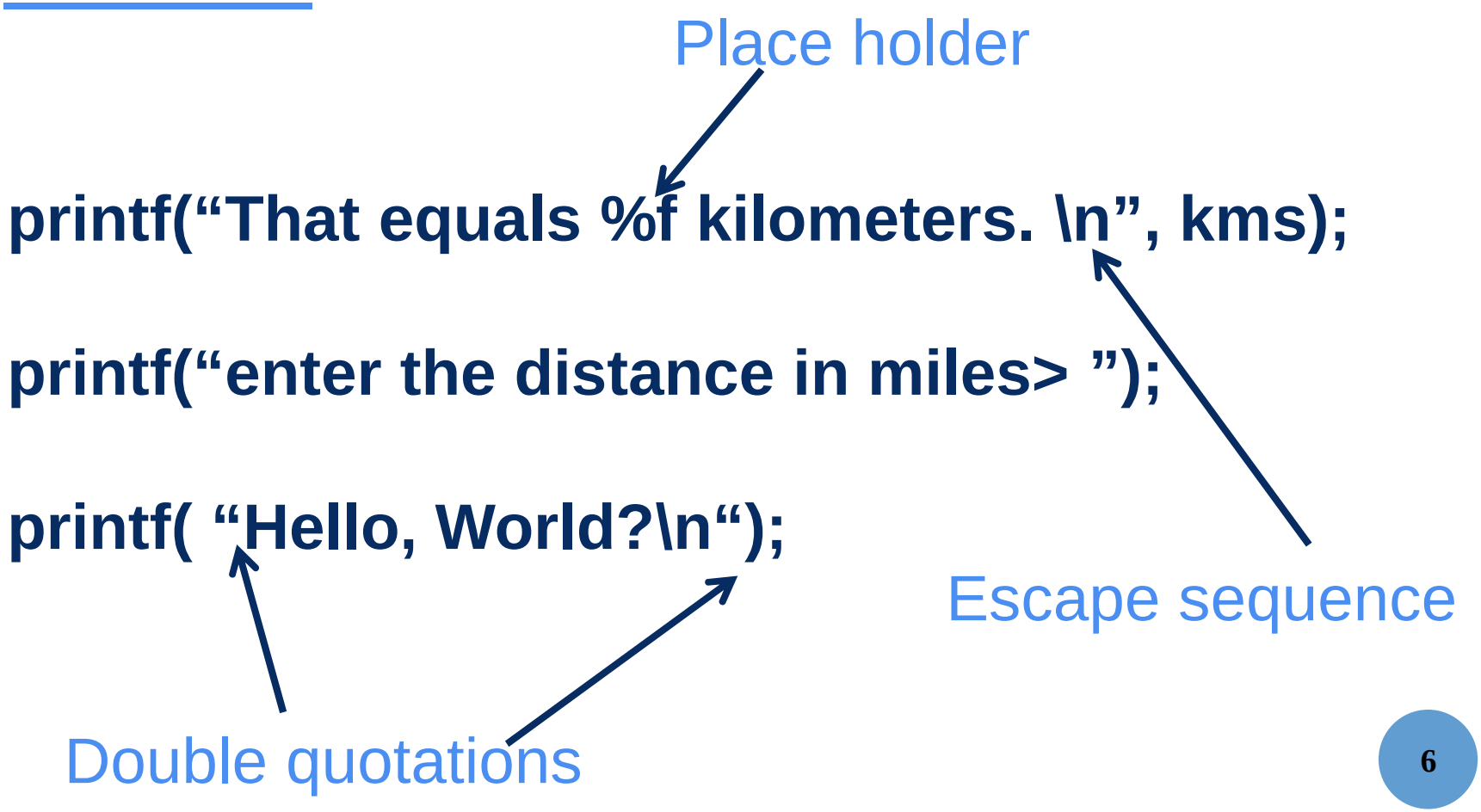
```
printf( format string , print list ) ;
printf(format string);
```

OUTPUT FUNCTION



I. INPUT/OUTPUT OPERATIONS FUNCTION

EXAMPLE



Placeholders

A placeholder is a symbol beginning with % in a format string that indicates where to display the output value

Placeholder	Variable type
%c	char
%d	int
%f	float

Escape sequence

- **The backslash (\) is called an escape character.**
- **Indicates that *printf* is supposed to do something unusual.**
- **When encountering a backslash, *printf* looks to the next character and combines it with the backslash to form an *escape sequence.***

Escape sequence	Action
\n	Starts new line
\t	tab

EXAMPLE

```
printf("This is one line \n");
printf(" and \n this \t is \t another \n ");
```

```
This is one line
and
this    is      another
```

Input Function (scanf)

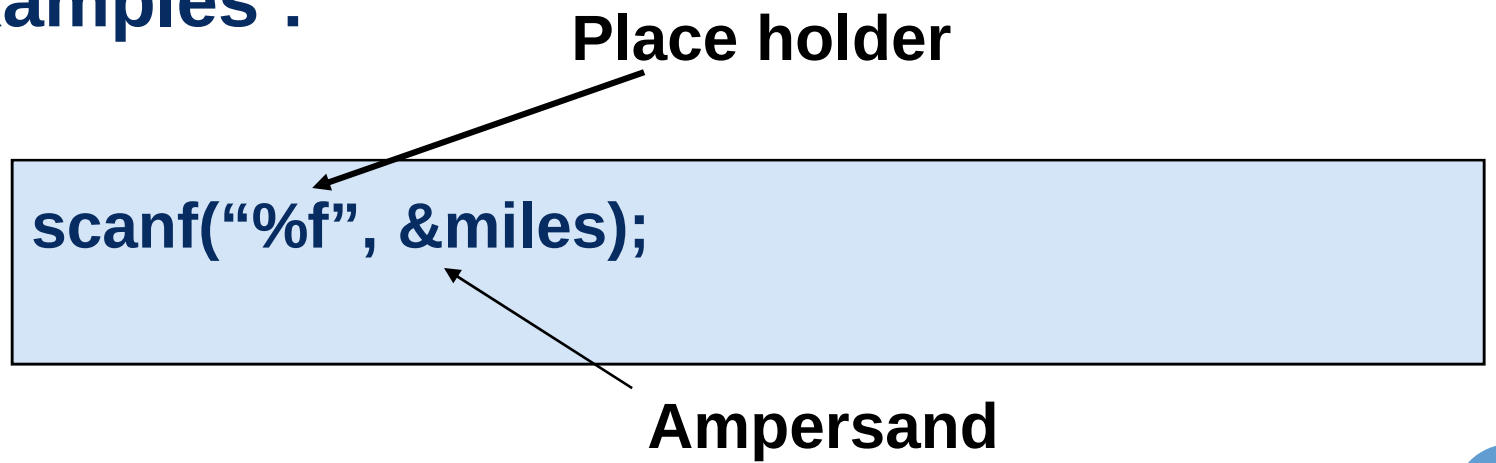
- The *scanf* function copies into memory data entered during the program execution.
- The order of the placeholders must correspond to the order of the variables in the input list.
- The data must be entered in the same order in the input list.
- You should insert one or more blank characters or carriage returns between numeric items.

INPUT FUNCTION

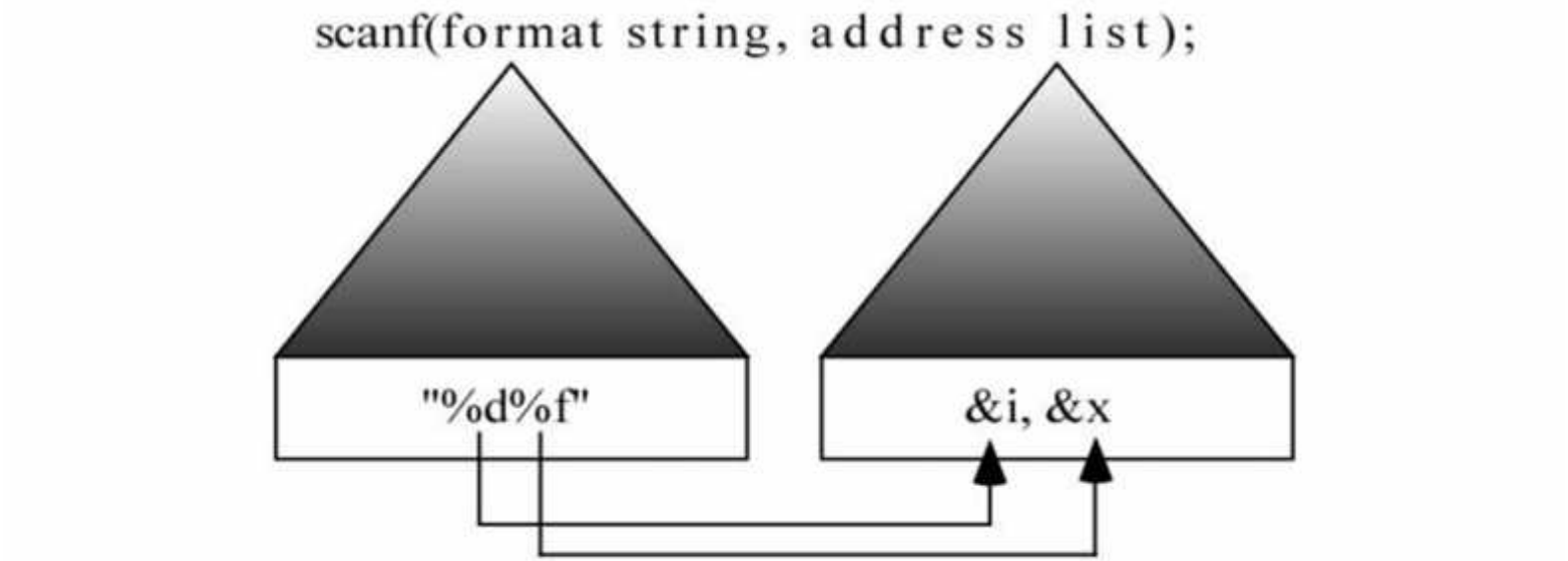
SYNTAX

```
scanf( format string , input list ) ;
```

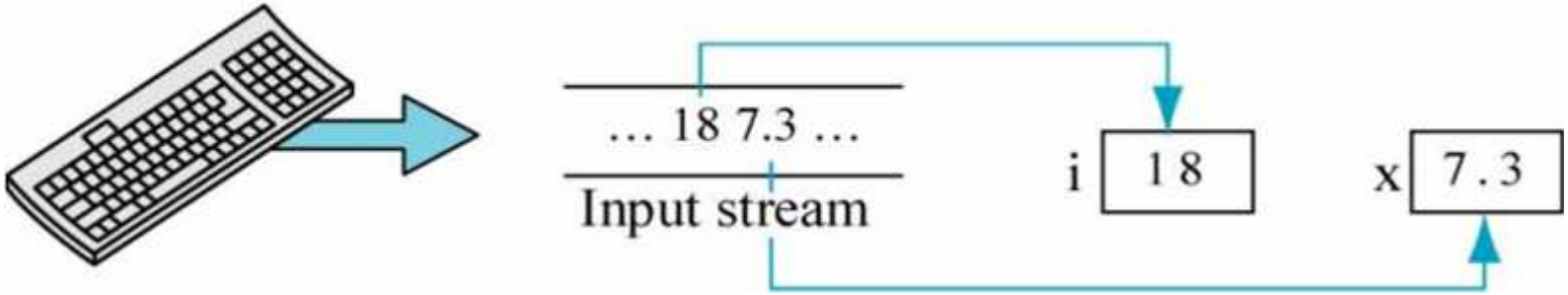
Examples :



INPUT FUNCTION



scanf("%d%f", &i, &x);



Output Formatting

- **In C, you can control the APPEARANCE of number on the screen.**

- **Output formatting does not affect the VALUE of a variable, just how it appears to the user on the screen.**

Output Formatting
Integer formatting

- **Very simple: Add a number between the % and the d in the placeholder to specify the “field length”.**

- **Numbers will appear right-justified with preceding blanks if needed.**

INTEGER FORMATTING

EXAMPLE

```
int len = 234 ;
printf(" Length is %5d ", len);
```

Output is:

```
Length is  △△234
```

Note: The △ stands for a blank

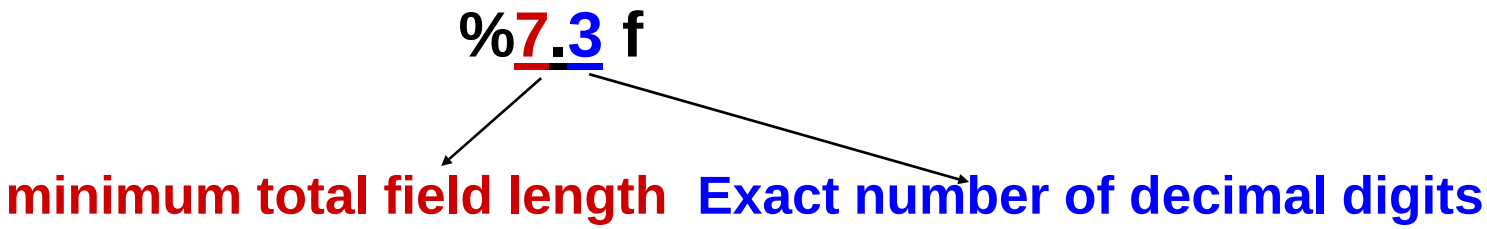
INTEGER FORMATTING
EXAMPLE

Value	Format	Displayed Output	Value	Format	Displayed Output
234	%4d	△234	-234	%4d	-234
234	%5d	△△234	-234	%5d	△-234
234	%6d	△△△234	-234	%6d	△△-234
234	%1d	234	-234	%2d	-234

OUTPUT FORMATTING

FLOAT FORMATTING

We must indicate both the field width and the EXACT number of decimal places:



Note: The decimal part will be rounded

The whole part may be padded with blanks

REMEMBER: The value of the number does not change, only its appearance

Double Formatting

Displaying X using different %6.2f placeholder

Value of x	Displayed Output	Value of x	Displayed output
-99.42	-99.42	-25.554	-25.55
0.123	△△0.12	99.999	100.00
-9.536	△-9.54	999.4	999.40

II. PROGRAMMING EXAMPLES

EXAMPLE 1:

Write a program to ask the user for the width and length of a piece of land and then tell him how many orange trees he can grow on it. Given that each orange tree requires 4 m^2.

EXAMPLE 1:

```
#include <stdio.h>
# define one_tree_space  4
void main()
{
    int   length,width, area, no_of_tree;
    printf(“Enter length of the land> ”);
   scanf(“%d”, &length);
   printf(“Enter width of the land> ”);
   scanf(“%d”, &width);
    area = length * width;
    no_of_tree = area / one_tree_space;
     printf(“The available number of trees is %d  trees\n”,
    no_of_tree);

}
```

EXAMPLE 2:

Write a program to ask the user for the radius of a circle, and then display its area and circumference, displayed to 3 decimal digits.

EXAMPLE 2:

```
#include <stdio.h>
# define PI  3.141593
int main(void)
{
    double  radius, area, circumference;
    printf("Enterradius of the circle> ");
   scanf("%lf", &radius);
    area =  PI * radius * radius;
    circumference = 2 * PI * radius;
    printf("The area of the circle =  %.3f\n", area);
 printf("The circumference of the circle =  %.3f\n",
    circumference);
     return(0);

}
```

EXAMPLE 3:

Write a program to create two integer variables, and store in them the values 3 and 5. Then calculate their sum and their product and display the result . Use comments.

EXAMPLE 3: *SAMPLE 1*

```
/*
*  This program gets the sum and product of 2 integers
*/
#include <stdio.h>                          /* library containing printf  */
int main (  )
{
            int first ;                     /* declaring first variable */
            int second ;                    /* declaring second variable */
            int sum ;                       /* variable to hold the sum */
            int product ;                   /* variable to hold the product */
   first = 3 ;                               /* assignment statement */
  second = 5 ;
  sum = first + second ;
  printf(" The sum is %d  \n ", sum );  /* output */
  product = first * second ;
  printf(" The product is %d \n ", product );
  return  (0);
}
```

EXAMPLE 3: *SAMPLE 2*

```
/*
*  This program gets the sum and product of 2 integers
*/
#include <stdio.h>                    /* library containing printf  */
int main (  )
{
    int first = 3 ;              /* declaring and assigning first variable */
    int second = 5 ;             /* declaring and assigning second variable */
     int sum, product ;

     sum = fiirst + second ;
     product = first * second ;

     printf(“ The sum is %d \n The product is %d \n “, sum, product );

     return ( 0 ) ;
}
```

THANK YOU