

Lab Manual

Course : CSE -105

Credit Title : Structured Programming

Instructor : Md.Shamsujjoha (MSJ)

Lab-2: Inputs, Output and Conditional Statements

Part 1: Inputs, Output

Aims: By the end of the workshop you should be able to understand: -

- Detail of C's `scanf` and `printf` functions and How to take input from keyboard through `scanf` ? also How to solve a general problem with C ?

In the previous lab we go through the *printf* in detail. So, we can start with *scanf*. As discussed in the class *scanf* is the C library's counterpart to *printf*. Actually, *scanf* requires a format string to specify the appearance of the input data. Example of using *scanf* to read an int value:

```
scanf("%d", &i);
```

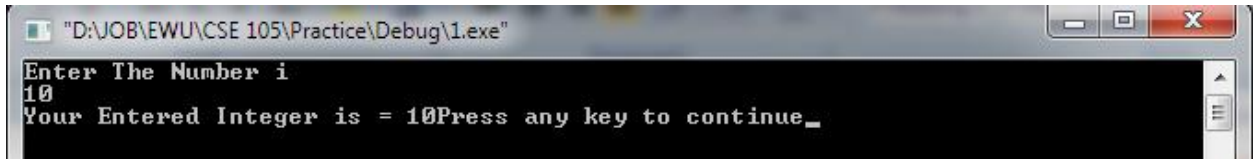
The above code will read an integer and stores into variable `i`. The `&` symbol is usually (but not always) required when using *scanf*. Now write the following code:

```
#include <stdio.h>
int main()
{
    int i;
    printf("Enter The Number i\n");
    scanf("%d",&i);
    printf("Your Entered Integer is = %d",i);
    return 0;
}
```

1. Compile and execute the above code.
2. When you execute the code you will certainly see a screen like this.



3. If not then there is obviously something wrong in your code, in that case knock your teacher.
4. If there is nothing wrong please enter **10** form your keyboard and press enter.
5. Your output should be as follows, otherwise knock your teacher.



```

D:\JOB\EWU\CSE 105\Practice\Debug\1.exe
Enter The Number i
10
Your Entered Integer is = 10
Press any key to continue_

```

6. Now execute the code again and take **10.50** form your keyboard.
7. The output should be the same as in step 5. In output **.50** is missing, this is because we take the variable *i* as an integer. If we want to print the exact **10.50**, we should declare variable *i* as **float or double**.
8. Reading a float value requires a slightly different call of *scanf*:
scanf("%f", &i);
"%f" tells scanf to look for an input value in float format *i.e.* the number may contain a decimal point, but doesn't always have to.
9. If you declare *i* as double your *scanf* should be as follows:
scanf("%lf", &i);
"%lf" tells scanf to look for an input value in double format.
10. For the *printf* you have to use %d or %f or %lf respectively. Look at table 1 for more detail. Don't worry about the **char and string** we will discuss more detail in the upcoming labs.

	printf	scanf
int	%d	%d
float	%f	%f
double	%f	%lf
char	%c	%c
string	%s	%s

11. If you want to enter more than one value *i.e.*, serialized the inputs you can do it by the following code.

```

float height, weight;
.....

scanf("%f%f", &height, &weight);

```

Exercise 1 : In this initial exercise you are asked to calculate the volume of a room. As you know $volume = width \times height \times length$. In the C this formula will be $volume = height * length * width$;

That is operation $*$ in C is same as \times in mathematics. For more operators in C look at following table

- Addition + `sum = num1 + num2;`
- Subtraction - `age = 2007 - my_birth_year;`
- Multiplication * `area = side1 * side2;`
- Division / `avg = total / number;`
- Modulus % `lastdigit = num % 10;`
 - Modulus returns remainder of division between two *integers*
 - Example `5%2` returns a value of 1
- Binary vs. Unary operators
 - All the above operators are binary (why)
 - - is an unary operator, e.g., `a = -3 * -4`

Your job is to declare width, height and length as double. Write the code to take these as input from keyboard. Print the volume.

Sample input	Sample Output
Enter height of box: 10 Enter length of box: 10 Enter width of box: 10	Volume (cubic inches): 1000
Enter height of box: 10 Enter length of box: 30.50 Enter width of box: 50.75	Volume (cubic inches): 15478.750000

Exercise 2 : Compute the straight line distance between two points in a plane. i.e. your job is to take the points as input from keyboard and print the outputs. The coordinates of points should be declared as float and for this you should know how to use *sqrt()* with *#include<math.h>* as discussed in the class.

Sample input	Sample Output
Enter Point X1: 1.0 Enter Point X2: 4.0 Enter Point Y1: 5.0 Enter Point Y2: 7.0	The distance is 3.605551
Enter Point X1: 11.0 Enter Point X2: 14.0 Enter Point Y1: 5.0 Enter Point Y2: 8.0	The distance is 4.242640

Part 2 : C Operators and its uses.

1. Declare two variable x and y as *int*
2. Assign a value to this variables by taking input from keyboard
3. Calculate and print
 - a. $a = x+y;$
 - b. $b = x*y ;$
 - c. $c = x/y;$
 - d. $d = x\%y;$
 - e. $e = x-y;$
4. What is the result when we declare **a to e as int ?**
5. What is the result when we declare **a to e as float ?**
6. What is the result when we declare **a to e as double ?**
7. What is the result when we declare **a to e as int ?**
8. What is the result when we declare **a to e as int but in step 1 but x and y are float or double ?**
9. Do you know what is the reason behind this ?
10. Repeat step 1 to 9 with different combination such as, int, float and double.

Exercise 1 : Relation between Celsius and Fahrenheit is governed by the formula: $F=9*C/5+32$. Write a program to convert the temperature from Fahrenheit to Celsius.

Sample Input	Sample Output
130.5	54.72
-20.25	-29.03

Exercise 2 : You are given two integer numbers. Find the quotient and remainder.

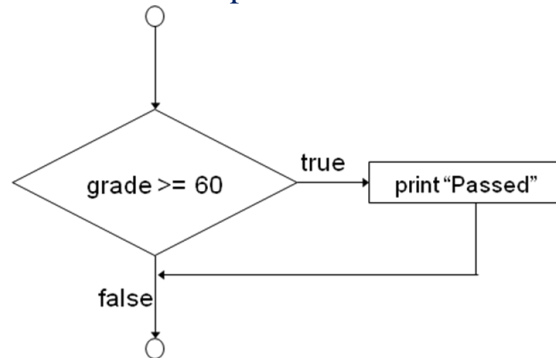
Sample Input	Sample Output
17 5	Q = 3, R= 2

Exercise 3 : Write a program that takes 3 digit number as input, reverse the number and print the reversed number.

Sample Input	Sample Output
123	321
342	243

Part 3 : Conditional Statements: *if, if-else, if-else-if*

As discussed in the class Selection structure are used to choose among alternative courses of action: In this problem we want to check whether a student is passes or failed. According to the rule of EWU If as students marks is greater than or equal to 60 he/she “Passed”. Flowcharts of this problem is as follows :



In C we can code this as follows

```
if(grade>=60)
    printf( "Passed\n" );
```

Exercise 1 : Now write the following code

```
#include<stdio.h>
#include<math.h>

int main()
{
    int grade_number;
    printf("Enter The Number\n");

    scanf(" %d",&grade_number);

    if(grade_number>=60)
        printf("\nPassed\n");

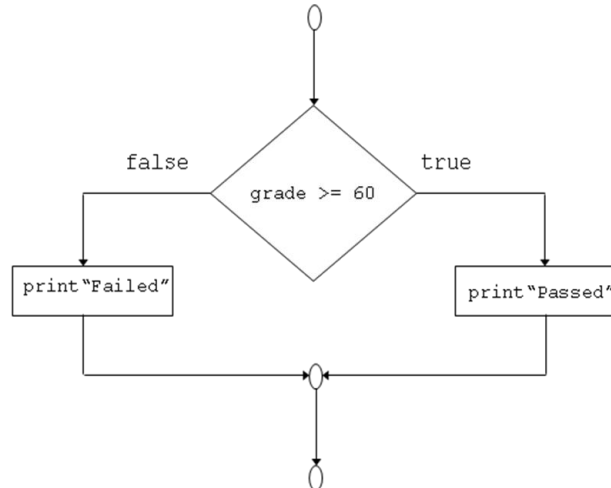
    return 0;
}
```

1. Fill the printf and scanf line such that the output would be as follows.

```
Enter The Number
70
Passed
Press any key to continue
```

```
Enter The Number
59
Press any key to continue
```

- As you see there is no output for the input 59 .This is because *if* statement is a single-entry/single-exit structure, i.e. our code will perform action for the true condition only.
- Using *else* we can convert the above code such it can perform for the false condition too. In that case flow chart will be as follows



- In the above code replace the *if parts* with the following code .

```

if (grade_number >= 60)
    printf ( "\nPassed\n" );
else
    printf ( "\nFailed\n" );

```

- What is the output for the input **70** and **59** here?

Note : In the *if* statement template, notice that statement is singular, not plural:

```

if(expression)
    statement

```

To make an *if* statement control two or more statements, use a compound statement. A compound statement has the form { statements } Putting braces around a group of statements forces the compiler to treat it as a single statement i.e. C code will be as follows:

```

if(expression)
{
    Statement_1;
    Statement_2;
    ...
    Statement_n;
}

```

Exercise 2 : Modify your code such that it can follow the following output

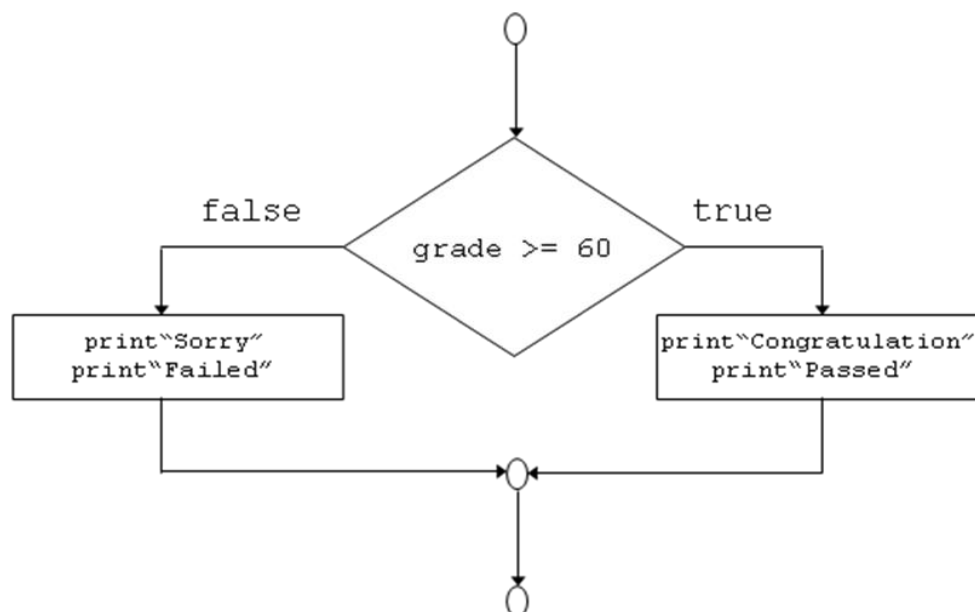
```
Enter The Number
70

Congatulation
Passed
Press any key to continue_
```

```
Enter The Number
50

Sorry
Failed
Press any key to continue_
```

That corresponding flow chart will be as follows:



For this you should use separate *printf* for each text i.e., one *printf* for Sorry, one for Failed, one for Congratulations and another one for Passed .

Exercise 3 : This is the repetition of Exercise 2 of part 2. You are given two integer numbers. Find the quotient and remainder. Now you should code with proper error checking, such as when 2nd input is zero you should print “Error!! Can’t divide.”

Sample Input		Sample Output
17	5	Q=3, R=2
3	0	Error!! Can’t divide.

Home Works:

1. Write program that will take one integer number as input representing number of second. Convert this number to the corresponding hour, minute and second.

Sample Input	Sample Output
106	0 Hour, 1 Minute, 46 Second
43239	12 Hour, 0 Minute, 39 Second

2. Suppose you are given 4 integer numbers “a”, “b”, “c” and “d” representing two rational numbers: a/b and c/d. Perform addition, subtraction, multiplication and division operation for these two rational numbers.

Sample Input	Sample Output
6 7 2 3	Addition: 32/21 Subtraction: 4/21 Multiplication: 12/21 Division: 18/14

3. Write a program that will take an integer input representing year and print whether this is a leap year or not leap year.

Sample Input	Sample Output
2000	Leap Year
2003	Not Leap Year

4. The line joining the points (a, b) and (x, y) which lie on the circumference of a circle is the diameter of the circle. Write a program to compute the area of the circle. *a*, *b*, *x* and *y* are given as input.

Sample Input	Sample Output
4 10 1 5	26.70

5. In the class we have discussed a lot about the C's relational operators i.e. about +, -, *, /, %. Now write a C code that can check whether a number is odd or Even. Your number should be input from the keyboard. If the number is odd your programme should print “The number you entered is ODD” otherwise it should print “The number you entered is EVEN”

Sample input	Sample Output
Please Enter your number : 11	The number you entered is ODD
Please Enter your number : 120	The number you entered is EVEN

6. Write a C programme to compute the value of the following functions, where x is the input from keyboard.

$$f = \frac{x^3 - 2x^2 + x - 6.3}{x^2 + 0.05x + 3.14}$$

Sample input	Sample Output
Please Enter the value of X : 4.75	The value of the function is 2.332185
Please Enter the value of X : 11	The value of the function is 8.771353

7. Write a C programme to compute the area of a circle, where the radius is the input from your keyboard. If the area of the circle is less than 100 square units then your programme should print “The Circle Is Too Small To Hold A Quadrate”, otherwise it should print “Your Circle Is Big Enough To Hold A Quadrate & The Area of The Quadrate is **X**”. Here **X** is the actual area of the Quadrate.

Sample input	Sample Output
Please Enter The Radius : 5.5	The Circle Is Too Small To Hold A Quadrate
Please Enter The Radius : 6.75	Your Circle Is Big Enough To Hold A Quadrate & The Area of The Quadrate is 91.125000

Reminder: Next week no student will be allowed to enter into the lab without completion of the above home tasks. Submission of the home tasks can be either Hard Copy (Hand Written or, printed) or Soft Copy (Source Code). We also *vigorously* opposed to the academic dishonesties, as it seriously detracts from the education of honest students. In the completion of your home works, it is impermissible to discuss a general method of solution with other students, or to make use of online resources. In case of any problem, inconsistency and errors in coding knock your Instructors/teachers. We are here to help you and to remove your confusions.

Thanks

Maheen Islam, Dr. Taskeed Jabid and Md. Shamsujjoha

Faculty members, Department of Computer Science & Engineering, East West University

Email: maheen@ewubd.edu, tasked@ewubd.edu and dishacse@yahoo.com