Name: _____ Student ID: _____ Date: _____

Worksheet 4

Basic Input functions and Mathematical Operators

Objectives

After completing this worksheet, you should be able to

- Use an input function in C
- Declare variables with suitable data types
- Understand the conversion specifications in an input function

1. Open a new project and write Program 4.1, then run it.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
        int x;
        float y;
        double z;
        printf("Please enter an integer value :");
        scanf("%d",&x);          // Wait for an integer value from a user
        printf("Please enter a float value :");
        scanf("%f",&y);          // Wait for a floating point value from a user
        printf("Please enter a double value :");
        scanf("%lf",&z);         // Wait for a double precision floating point value
        printf("\n\n");          the letter "l" (L)
        printf("Your integer value is %d \n",x);
        printf("Your float value is %f \n",y);
        printf("Your double value is %f \n",z);

        return  0;
}
```

Program 4.1

Enter the following values to the program

Please enter an integer value : **365**

Please enter a float value : **99.12** ← **Entered by a user**

Please enter a double precision value : **34567890.75**

Record the result produced by the program

2. Open a new project and write program 4.2.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{                    //To enter values to the program, users have to add
  int a, b;          //a space between two input values Ex. Enter 2 integer values : 32  94
printf("Enter 2 integer values : ");
scanf("%2d %4d",&a,&b);                  // Add one space (press the key spacebar)
printf("a = %d  b = %d",a,b);


return 0;
}
```

Program 4.2

3. Run the program 4.2 and input to the program the values in Table 4.1 and then record the result.

Input values	Results	
	a =	b =
32 94		
32 12345		
345 12345		

Table 4.1

4. Open a new project, write program 4.3 and run it.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
        int hour, min, sec;
        printf("Please enter time in hh:mm:ss : ");
        scanf("%d:%d:%d",&hour,&min,&sec);  //To enter input data to the program, users
        printf("Current time : %02d:%02d:%02d",hour,min,sec);
        return 0;
}
```

Program 4.3

Input the numbers "**1:30:08**" to the program.

Please enter string: **1:30:08** press **Enter**

Record the result. .....

5. Open a new project, write program 4.4 and run it.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
        char ch;
        printf("Please enter one character : ");                    ----------①
        ch = getchar();          //getchar() is defined in the library "stdio.h"  ----------②
        printf("\nch =  %c\n",ch);                                  ----------③
        return 0;
}
```

Program 4.4

6. Enter the characters in Table 4.2 into the program and record the result in Table 4.2.
7. From program 4.4, modify the statement ② and rerun it.

   ch = getchar(); ➔ ch = getche();

   Enter the characters in Table 4.2 into the program and record the result in Table 4.2.
8. From program 4.4, modify the statement ② and rerun it.

   ch = getchar(); ➔ ch = getch();

   Enter the characters in Table 4.2 into the program and record the result in Table 4.2.

Input Characters	Results from 6-8		
	step 6.	step 7.	step 8.
A	ch = ____	ch = ____	ch = ____
a	ch = ____	ch = ____	ch = ____
H	ch = ____	ch = ____	ch = ____
+	ch = ____	ch = ____	ch = ____
1	ch = ____	ch = ____	ch = ____
Key “Esc”	ch = ____	ch = ____	ch = ____

Table 4.2

9. What is the difference among these three functions: getchar(), getche() and getch()?

10. Open a new editor and write Program 4.5, then run it. Record the result in Table 4.3.

```
#include <stdio.h>
int main(int argc, char *argv[]) {
  int a, b;      float c, d;
  a = 35;     b = 12;     c = 185.73;    d = 94.781;
  printf("a + b = %d\n",a+b);
  printf("a - b = %d\n",a-b);
  printf("a * b = %d\n",a*b);
  printf("a / b = %d\n",a/b);   /* an integer value /an integer value produces an integer value */
  printf("a % b = %d\n",a%b);
  printf("c + d = %f\n",c+d);
  printf("c - d = %f\n",c-d);
  printf("c * d = %f\n",c*d);
  printf("c / d = %f\n",c/d);  /* a floating point /a floating point value produces a floating  point value */
  printf("a + c = %f\n",a+c);
  printf("c / b = %f\n",c/b);       /* a floating point value /an integer value, and vice versa, produces */
                                    // a floating  point value
  return  0;
}
```

Program 4.5

Expressions	Results
	a = 35, b = 12, c = 185.73, d = 94.781
a + b	
a - b	
a * b	
a / b	
a % b	
c + d	
c - d	
c * d	
c / d	
a + c	
c / b	

Table 4.3

11. Open a new project and write program 4.6.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc,  char *argv[] )
{
        float Area, Height, Base;
        Height = 5.5;       Base = 10.0;
        Area = (1/2) * Height * Base;                    ----------①
        printf("Area of Triangle = %f",Area);

        return 0;
}
```

Program 4.6

12. Run the program 4.6 and record the results.

Area of Triangle = ...

Is the result correct? How can we correct the result?

13. Fill in the blanks with the correct answer.

- 25 / 6 = _____
- 25 / 6.0 = _____
- 3 / 12 = _____
- 15 % 5 = _____
- 8 % 12 = _____

Name: _____ Student ID: _____

Homework 4

1. Write a program in C that receives the information of three employees from the keyboard. The information of each employee is composed of **gender (m** or **f)**, **age**, **salary** and **telephone number**. Then display the information of three employees on the screen in appropriate form.

2. Given a set of mathematical expressions below,

2.1 $((x * y) + 3) - (7 + z)$, 2.2 $3x^2 + 2y - 7z$,

Write a program that accepts the values of variables **x**, **y** and **z** (as input variables) from the keyboard. Then calculate the expressions above by using these input variables. (Given that all of the variables used in this program are floating point variables).