AL-KAWTHAR UNIVERSITY

Department of Computer Science

CS 121 L – Programming Fundamentals (PF)

Lab # 02

Objective:

To introduce students to input/output functions (printf() and scanf()), data types, and format specifiers in C. Students will practice modifying code to reinforce understanding.

Name of Student	**Muhammad Hashir Rafique**
Student ID	**BSCS24101023**
Date of Lab Conducted	
Marks Obtained	
Remarks	
Signature	

LAB 2 - ACTIVITY 1

Basic Input/Output with printf() and scanf()

Objective:

- Use printf() to display output and scanf() to read input.
- Modify given code to handle different data types.

1. Sample Code:

```
#include <stdio.h>

int main() {
      int num;
      printf("Enter an integer: ");
      scanf("%d", &num);
      printf("You entered: %d\n", num);
      return 0;
}
```

2. Modification Tasks:

- Change the code to accept a **float** instead of an integer.
- Add a second scanf() to take a **character** input and print it.
- Explain why & is used in scanf() but not in printf().

Modified Code

```
#include <stdio.h>
int main() {
    float num;
    char letter;

    printf("Enter a float value: ");
    scanf("%f", &num);

    printf("Enter a character: ");
    scanf(" %c", &letter);

    printf("You entered the float: %.2f\n", num);
    printf("You entered the character: %c\n", letter);

    return 0;     }
```

- Accept a `float` instead of an integer.
- Add a `char` input and display it.
- `scanf()` needs the memory address of the variable to store the input value. The & (address-of) operator gives this address. While `printf()` only reads the value to display, so it does not need the memory address.

LAB 2 - ACTIVITY 2

Format Specifiers and Data Types

Objective:

- Practice using correct format specifiers (%d, %f, %c, %s).
- Observe errors caused by mismatched specifiers.

1. Sample Code:

```
#include <stdio.h>

int main() {
    float price;
    printf("Enter price: ");
    scanf("%f", &price);
    printf("Price: %.2f\n", price);
    return 0;
}
```

2. Modification Tasks:

- Modify the code to use %d instead of %f. What happens?
- Add a char variable to store a grade (A/B/C) and print it with %c.
- Use %.2f to print the price with 2 decimal places.

Modified Code

```
#include <stdio.h>

int main() {
    float price;
    char grade;

    printf("Enter price: ");
    scanf("%f", &price);

    printf("Enter grade (A/B/C): ");
    scanf(" %c", &grade);

    printf("Price (two decimals): %.2f\n", price);  // Task 2-c
    printf("Grade: %c\n", grade);

    return 0;
}
```

- Program runs, but when a variable value is stored (either decimal or floating) it returns (garbage value) 0.
- Add a char variable to store a grade (A/B/C) and print it with %c. --- Done
- Use %.2f to print the price with 2 decimal places. --- Already done

LAB 2 - ACTIVITY 2

Advanced Input Handling

- Handle multiple inputs in one scanf().
- Use escape sequences (\n, \t) for formatting.

1. Sample Code:

```
#include <stdio.h>

int main() {
    char name[20];
    int age;

    printf("Enter name and age: ");
    scanf("%s %d", name, &age);

    printf("Name: %s \t Age: %d\n", name, age);
    return 0;
}
```

2. Modification Tasks:

- Add a third input for height (float) and print it.
- Replace \t with \n and observe the output change.
- Explain why & is not used with name in scanf().

Modified Code

```
#include <stdio.h>

int main() {
    char name[20];
    int age;
    float height;

    printf("Enter name, age, and height: ");
    scanf("%s %d %f", name, &age, &height);

    printf("Name: %s\nAge: %d\nHeight: %.2f\n", name, age, height);
    return 0;
}
```

- Add a third input for height (float) and print it. --- Done
- Now it does not gives a tab (4 spaces) it enters a newline.
- The operator & is not used with name in scanf(), because we already declared an array of characters. Since array is already address of the characters we do not have to write &.

LAB 2 - ASSIGNMENT

Prepare a document that includes the following:

1. Define printf() and scanf() with syntax examples.
2. Create a table of common data types (int, float, char, double) and their format specifiers.
3. Write a program to take a student's name, roll number, and CGPA as input and display them in a formatted table.
4. Draw a flowchart for the program in Activity 3.

---

(1)

`printf()` - Displays output on the screen.

Syntax:

```
printf("format string", variable);
```

Example:

```
int age = 20;

printf("Age is %d", age);
```

`scanf()` - Takes input from the user.

Syntax:

```
scanf("format string", &variable);
```

Example:

```
int age;
scanf("%d", &age);
```

(2)

Data Type	Keyword	Format Specifier
Integer	`int`	`%d`
Float	`float`	`%f`
Character	`char`	`%c`
Double	`double`	`%lf`
String	`char[]`	`%s`

(3)

```c
#include <stdio.h>

int main() {
    char name[30];
    int rollNo;
    float cgpa;

    printf("Enter student name: ");
    scanf("%s", name);

    printf("Enter roll number: ");
    scanf("%d", &rollNo);

    printf("Enter CGPA: ");
    scanf("%f", &cgpa);

    // Display using tabs (\t)
    printf("\nStudent Details:\n");
    printf("Name\t\tRoll No\tCGPA\n");
    printf("%s\t\t%d\t%.2f\n", name, rollNo, cgpa);

    return 0;
}
```

FLOWCHART FOR ACTIVITY 3