GJU
الجامعة الألمانية الأردنية
German Jordanian University

CS1160
Lab assignment 2: Variables

Date 18 October 2020 **Online Lab Session**

I. Overview

In this lab you will learn about variables, how to define them and assign values to them. You will also learn how to use variables in simple calculations.

II. Variables declaration

A variable is a container (storage place) that has memory allocated to it and is used to store various types of data. Variables must be defined before they can be used within a program. Variable declarations include the type of the variable and –optionally- its initial value.

- A typical variable declaration is of the form:

```
type variable_name;
type variable_name = value;
```

to define multiple variables at once:

```
type variable1_name, variable2_name, variable3_name;
```

- The most common variable types used in C:
    - char: used for declaring character type variables. For example:

```
char name = 'B';
```

- int: used for declaring an integer variable. For example:

```
int year = 2020;
```

- float and double: used for declaring real numbers. The difference between float and double is the allocated memory size; size of float is 4 bytes, and the size of double is 8 bytes. Example:

```
float salary = 753.96;
double factor = 99.873;
```

- Rules for defining variables:
  - A variable name can have alphabets, digits, and underscore.
  - A variable name can start with the alphabet, and underscore only. It can't start with a digit.
  - A variable name is case sensitive; the upper and lowercase case letters are different.
  - No space is allowed within the variable name.
  - A variable name must not be any reserved word or keyword (e.g. void, int, for)

III. Reading and printing variables

- The function printf is used to send formatted output to the screen. Example:

```
printf("Enter your number: ");
```

- The function scanf takes formatted input from the user.

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

- **Specifiers:** The format specifiers are used in C for input and output purposes, so the compiler can understand what type of data is in a variable.

Format Specifier	Type
%c	Character
%d	Integer
%f	Float
%lf	Double

```
printf("your number is:", %d);
```

IV. Operators

An operator is a symbol that tells the compiler to perform specific mathematical or logical functions.

- Arithmetic Operators: + , - , * , / , % , ++ , --
- Relational Operators: > , < , >= , <= , == , !=
- Logical Operators: && , || , !

V. Example

```c
#include <stdio.h>
int main()
{
    int FirstInteger;
    int SecondInteger;
    char name;



    printf("Enter the first integer: ");
    scanf("%d", &FirstInteger);

    printf("Enter the second integer: ");
    scanf("%d", &SecondInteger);

    printf("Enter your name: ");
    scanf("%c", &name);


    int sum = FirstInteger + SecondInteger;
    printf("The sum is = %d",sum);
    printf("Your name is is = %c",name);

    return 0;
}
```

VI. Tasks

1. Write a programme to read and print your phone number, your initials and your average.
2. Write a programme to find the average of 3 numbers.
3. Write a programme that calculates your gpa.