

Solved Exercises

# Computer Science Grade 10th

According to academic year 2023-24 KP Text  
Book Board

By: IQBAL AHMAD SST-IT

BCS - UNIVERSITY OF PESHAWAR

ANDROID APP DEVELOPMENT - KPITB | Directorate of IT PESHAWAR



For Feedback:

Reach at [iqbalahmaddg@gmail.com](mailto:iqbalahmaddg@gmail.com)

[www.iqbalahmadcs.com](http://www.iqbalahmadcs.com)

**Q2. Give short answers to the following questions.****i. Define problem analysis.**

A computer does not have the intelligence to understand a problem and to prepare solution for it. So a programmer analyzes the problem then prepares instructions for its solution and gives it to the computer to solve the problem.

Technically problem analysis is studying the problem to know how it occurred and reached the current state. It finds the root causes of the problem.

Analyzing the problem involves the following factors:

- A clear idea about the problem and a description to state what is the problem about.
- Where and how frequently does it occur
- And who is effected by the problem.

**ii. How the solution of the problem is planned?**

The solution of the problem is planned after its full analysis. Normally there are different solutions exist for a problem but we will choose the solution which is less costly, take less time and easy to develop and implement.

**iii. Define candid solution of a problem?**

Among all the possible solutions to problem, the one which is “the best” is known as candid solution. It is efficient, simple and less costly.

**iv. Define any three problem solving techniques?**

Some problem solving techniques are as under:

**Abstraction:** to solve the problem in a model of the real system before applying to real one.

**Divide and Conquer:** to break down a large problem into small problems and solve them.

**Brainstorming:** a group of people sit and think to suggest various solution to the problem and continue until they reach the solution.

**v. List various factors for selecting the best solution to the problem.**

The following factors are taken into focus to select the best solution:

**Speed:** time is a big factor. The solution that takes less time i.e. speedy will be best.

**Cost:** we will consider the solution better if it gives good result using less hardware and software resources. i.e. less costly.

**Complexity:** the solution which performs better using little amount of storage space and with little amount of CPU execution time is known to be best.

\*\*\*\*\*

**Q3. Define an algorithm and explain the role of algorithm in problem solving.**

**Ans:** The list of well-defined steps to solve a problem is called **Algorithm**. algorithm takes some input value, applies computational steps on it for processing and then produces an output. In other words, algorithm transforms input into output.

**Role of Algorithm in problem solving:**

Algorithm plays an important role in developing solution to a problem. During solving complex problems that need a lot of input data and processing, algorithm divides the complex problem into unique steps and solve each step in sequence due to which the problem solution becomes an easy task. This property of algorithm is referred to as “Divide and Conquer”.

For example, if we have to find average of 20 numbers, the algorithm steps will be as below:

1. Get 20 numbers
2. Add all the numbers(sum)
3. Divide the sum of step 2 by 20
4. The result is ready.

**Q4. Describe the criteria for measuring efficiency of an algorithm on the basis of:**

**a. Inputs Needed B. Processing to be completed. C. decision to be taken d. outputs to be provided.**

**Ans:**

**Input Needed:** if the size of input data is large, the algorithm will take long to operate on it. For example, multiplying two numbers will take little time than multiplying two matrices.

**Processing to be completed:** the larger is the instruction set, the longer will it take the algorithm to complete the process. An efficient algorithm will complete large processing in little time.

**Decision to be taken:** Decision making or conditional operations are the factor that effect the efficiency of an algorithm. Better logic will result in better performance.

**Outputs to be Provided:** An algorithm will be correct if it provides correct output for each input. An algorithm will be considered best solution to a problem if it provides desired output for all of its inputs.

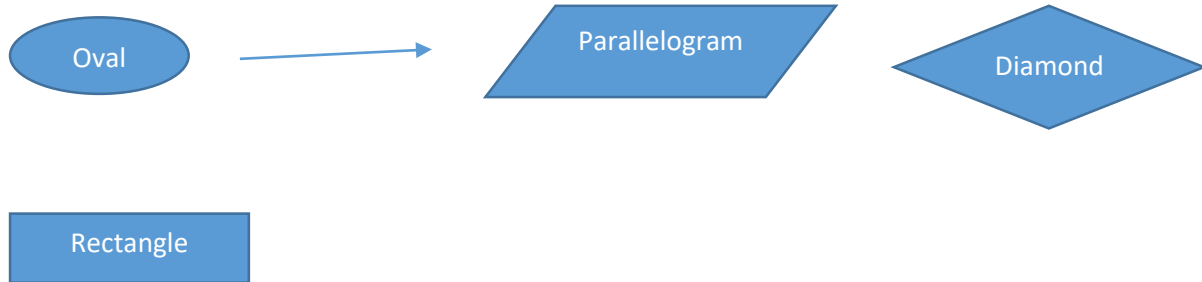
**Q5. What is flow chart, explain flow charts symbols in detail.**

**Ans:** The diagrammatic representation of an algorithm is called flowchart. The symbols of the flowchart contain information which represent a step or process of the algorithm of a problem solution. The symbols are connected through arrows which show the step by step flow of the process. Flow charts are commonly used in developing business plans, designing algorithms and determine trouble shooting steps.

**Flow chart symbols:**

Flow chart use special shapes to represent different actions or steps of an algorithm, these are called flowchart symbols. Some flow chart symbols are below:

1. **Oval:** Represent start or end of a flowchart.
2. **Arrow lines:** they are used to connected steps together in the direction of flow.
3. **Parallelogram:** it is used to represent input or out.
4. **Rectangle:** it represents computational step or process.
5. **Diamond:** it is used to represent a logical step, to make a decision or to answer a question wit Yes or No.



**Q6. Write an algorithm to calculate the factorial of a given number and draw flow chart for it.**

**Algorithm**

**Step 1: Read N**

**Step 2: set F = 1 and K = 1**

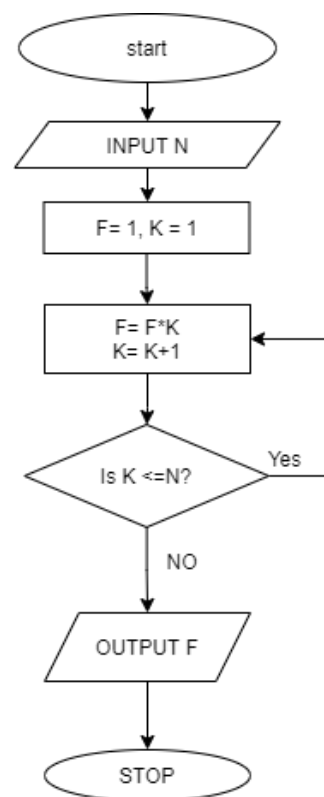
**Step 3: set F=F\*K**

**Step 4: K=K +1 [ increment K]**

**Step 5: if k<=N got step 3**

**Step 6: Write F**

**Step7: Exit.**



**Notes Prepared by: IQBAL AHMAD (SST-IT)**  
**[BS - Computer Science]**

**Note: Errors, Improvements, suggestions and Feedback about the notes will be welcomed at:**  
[iqbalahmaddg@gmail.com](mailto:iqbalahmaddg@gmail.com)

Q7. Write an algorithm and draw a flowchart to resolve quadratic equation.

**Algorithm**

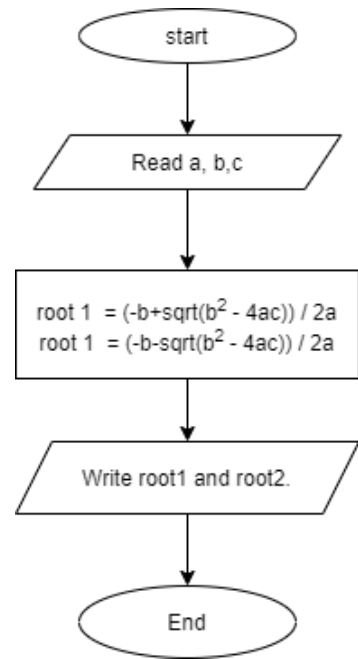
**Step 1:** Read a,b,c (coefficients)

**Step 2:** set root1 =  $(-b + \sqrt{b^2 - 4ac}) / 2a$

**Step 3:** set root 2 =  $(-b - \sqrt{b^2 - 4ac}) / 2a$

**Step 4:** Write root1 and root 2

**Step5:** Exit.



Q8. Write an algorithm and draw a flowchart to find average marks of eight subjects for student.

**Algorithm**

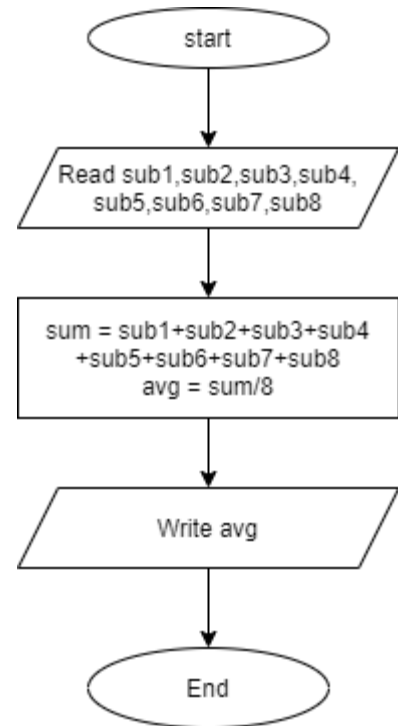
**Step 1:** Read s1,s2,s3,s4,s5,s6,s7,s8

**Step 2:** set sum = s1+s2+s3+s4+s5+s6+s7+s8

**Step 3:** avg = sum/8

**Step 4:** Write avg

**Step5:** Exit.



**Q2: Give short answers to the following questions.****i. Differentiate between program syntax and program semantics.**

The set of rules associated with a programming language according to which the statements of a program will be written is called program syntax. It tells us how to write correct statements of a program. It is similar to the grammar of any natural language.

E.g. Syntax of print statement in C is: ***printf ("hello world");***

Semantic is the meaning associated with a statement(s) in a program. It tells us what the statement will do when it is run.

e.g. ***product = x\*y;*** semantic of this statement is that it will multiply values of ***x*** and ***y*** and assign the result to the variable ***product***.

**ii. Differentiate between low level and high level languages.**

**Low level languages** are those which are near to computer hardware. Computer understands them easily. However, writing program in low level language is troublesome and requires a deep knowledge of internal computer hardware structure. It has other subtypes like Machine language and Assembly language.

**High Level language** are human understandable languages. They are English-like languages. We write program in English oriented words, like ***print, input, if, while*** etc. each language has own set of rules called **syntax** for writing programs in it. The compiler or translator then translates the program written in high level language into machine language for computer to understand and process.

**iii. What is IDE?**

The tool or software which combines all the tools that are used to write, compile and run a program is called an IDE (Integrated Development Environment). Normally an IDE consist of an editor, compiler Loader and a debugger.

**iv. What is OOP?**

Object oriented programming represent real world entities as objects inside program. Every Object has own set of data structures(attributes) and functions (Properties) just like the real entity have. Object once created can be reused which makes programming easy and manageable. Popular OOP programming languages are C++ and JAVA etc.

**v. What are the characteristics of high level languages?**

Characteristics of high level language are as under.

**Easy to Learn:** since high level language uses English like words therefore it is easy to be learned than low level language.

**Easy to Understand:** program in written in high level language by one programmer can easily be understood another program because of its English oriented syntax.

**Easy to write program:** large programs can easily be written in high level language easily because it uses short statement as compared to low level language.

**Easy to detect and remove Errors:** it is easy to find and remove errors from a program written in high level language. Compiler points out most of the errors.

**Built-in Library Functions:** each high level language provides a variety of libraries that include predefined functions for various tasks. These functions can easily be used in program source code.

**Machine Independence:** program written in high level language can be translated for any machine and OS. So it is likely to execute everywhere.

**vi. Differentiate between compiler and interpreter.**

**Compiler** is a system software that translates program written in high level into machine language object code. it translates (compiles) the whole program and then runs it.

**Interpreter** works just like compiler but it translates program statement by statement. it converts one statement into machine code, runs it and then proceeds to next statement.

**vii. What is header File?**

Header files in C is a collection of library functions that perform different tasks. There exist many header files in C and each contains different kind of functions. Header file is included in the program using **#include<header-file>**. Its extension is **.h**

**e.g.** header file input output functions are: **stdio.h**, for math **math.h**

**viii. Differentiate between source program and object program.**

Source program is the code statements that are written by a programmer. In C, the extension of a source program is **.c** or **.cpp**, however, when the program is compiled by the compiler it generate another file which called object program or object code. It is machine understandable format of the program its extension is **.obj**

**ix. What are reserved words?**

Those words which are used for some kind of special meaning in a language are called reserved words. They are also called keywords. They cannot be used for any other purpose in the program. For example, keywords in C are: **int, float, if, while, continue** etc.

**x. Write rules for variable names.**

Rules for writing variables in C are as under:

- Variable name must start with alphabet or underscore.
- Blank space not allowed in variable name.
- Variable name should not be a keyword.
- Only underscore as special character is allowed.
- One name should be declared once in a program. it cannot be re-declared.

**xi. What is purpose of const qualifier?**

The **const** qualifier / keyword is used to define a constant in the program. Constant is given a name and initialized with a value. Its value cannot be changed during program execution. It syntax is : **const data type identifier = value;**

**e.g.: const int N = 100;**

**Q.3 What is programming language? Explain different types of low level programming languages.**

**Ans:** Programming language is a set of words, symbols and codes that is used to write computer programs. There are many programming languages available and all of them are used to write computer programs to solve business and computational problems. Programming languages are mainly classified into two categories: **High level languages** and **low level languages**.

**Low level languages** are near to computer hardware. They are easily understood by the machine, however writing programs in low level languages is troublesome and requires immense knowledge.

Low level languages have two subtypes:

**Machine Language:** this language is understood directly by the machine. Programs in this language are written in binary format. Since it is the fundamental language of the computer therefore programs written in this language run very fast. Every machine (computer) has its own machine language syntax. It is difficult for human to understand and programming in machine language takes a lot of time.

**Assembly Language:** This language is one step higher than machine language. Symbols are used instead of binary for programming. These symbols are called **mnemonics**. For example, **sub** instruction is used for subtraction. This language is also called symbolic language and programs written in it are easy to modify than machine language. It is known as second generation language.

**Q.4 What are high level languages? Explain any five high level languages.**

**High Level language** are human understandable languages. They are English-like languages. We write program in English oriented words, like **print, input, if, while** etc. each language has own set of rules called **syntax** for writing programs in it. The compiler or translator then translates the program written in high level language into machine language for computer to understand and process. Some high level language are given below:

**C Language** was developed by Dennis M. Ritchie at bell labs in 1972. It is procedural language and has so many low level features like handling memory bit by bit. It runs on so many platforms. UNIX operating system was written in C.

**C++ Language** developed in early 1980 by Bjarne Stroustrup. It is an extension of C language with OOP features like working with objects, classes, events other object oriented concepts. Word processors and spread programs are developed in C++.

**Java Language** is a high level object oriented programming language. It was developed by sun microsystems in 1991. It is very similar to C++ however it visual support for developing web and desktop programs,

**C# Language** is also called C Sharp. It is in objected oriented programming language which was developed by Microsoft in 2000. It is based on C++ with other elements from visual basic and java. It is used to develop web as well as desktop and mobile applications like games, operating systems, compilers and other business applications.

**PHP:** It is an acronym for HYPER TEXT PREPROCESSOR. It is a server side scripting language designed specifically for web applications. It is open source which mean every can download and use it. It executes the code and the server which runs the backend logic for the websites and web applications.



**Q.5 What is the basic structure of a C-program? Also explain different types of preprocessor directives.**

A C program consists of the following three main parts:

1. **Preprocessor directives:**

These are the instructions given to the compiler before beginning of the compilation of a program. They are also called compiler directives. These directives are of two kinds:

- **#include** directives which are used to include header files for different functions. For example, **#include<conio.h>** for input and output, **#include<math.h>** for math functions etc.
- **#define** directives which allows the definition of identifiers or constants. For example **#define PI 3.14159**

2. **The main () function:**

The main function is a must part of every C program. Execution of a C program Starts form main() function. If it is not included, program will not compile.

3. **Body of main ():**

The statement written inside { } of the main() is called body of the program. Each statement ends with a semicolon. " ; "

**Q.6 What are constants and variables? Differentiate them with example.**

**A variable** is a memory reference whose value changes during program execution. The variable is represented by its name. name of variable remains unchanged but value can be changed. Variable may be numeric or character. Numeric variable stores numbers like age, roll number etc., while character variable stores alphabets and symbols like name, gender etc.

**A constant** is quantity which remains unchanged during programs execution. It will retain the initial value given to it in the start of the program.it can be used anywhere in the program. Same like variable, constant can be numeric or string(character)

**Q.7 Explain different data types used in C-Language with examples.**

The C language supports the following data types:

Integer(**int,long**), Floating point /decimal (**float**). Double precision float (**double**).

Character (**char**)

i. **Integer (int, long, unsigned int) Types:**

Integer data types stores whole numbers. The variants of integer in c language are as given:

Integer identifier	Memory bytes	Range
Int	2	-32,768 to + 32,767
Long	4	-2,147,483,648 to + 2,147,483,647
Unsigned int	2	0 to 65,535

e.g. **int age**

ii. **Floating point (float, double, long double) Types:**

float represents a real number(decimal point) up to 7 digits precision. Its other variants are **double** and **long double**. Details in the following table

Variable identifier	Memory bytes	Range
Float	4	$3.4 \times 10^{-38}$ to $3.4 \times 10^{38}$
Double	8	$1.7 \times 10^{-308}$ to $1.7 \times 10^{308}$
Long double 10	10	$3.4 \times 10^{-4932}$ to $3.4 \times 10^{4932}$

e.g. **float temperature;** **double product;**

### iii. Character (char ) type

char variable stores one character(alphabet or symbol). E.g. **char ch;** the C compiler allocates one byte of memory to the character variable.

### Q.8 What is type casting? Explain with examples.

The conversion of one data type into another data type during execution of the program is called type casting. It is performed in two ways:

implicit type casting

Explicit Type casting.

**Implicit Type casting:** implicit type casting is done automatically by the compiler. Suppose adding two variables, one is integer while the other is long, the lower(int) will automatically be converted into long.

**Explicit Type casting:** programmer is responsible for explicit type casting. Suppose finding modulus of a float variable, it is first to be casted to integer before operating otherwise compiler will through an error as we cannot find modulus of any float variable.

So this will be written like this: **(int) x % (int) y**

### Q.9 How variables are declared and initialized in C-Language? Give examples.

**Variable Declaration:** to name a variable along with its datatype is called variable declaration. Its syntax is: **data\_type variable\_name; e.g int age; float temperature; char ch;**

**int a,b,c;** (declaring many variables of same data type at once).

**Variable initialization:** When a variable is declared a location in memory is allocated for it. The value we assign to this variable later on will be stored in that location. However if we assign a value to the variable at the time of declaration ,this process is called initialization of the variable.

e.g. **int a= 10; int b=4,c=5; float n= 3.35;**

Notes Prepared by: IQBAL AHMAD (SST-IT) [BS - Computer Science]

Note: Errors, Improvements, suggestions and Feedback about the notes will be welcomed at:

[iqbalahmaddg@gmail.com](mailto:iqbalahmaddg@gmail.com)

[www.iqbalahmadcs.com](http://www.iqbalahmadcs.com)

**Q2: Give short answers to the following questions.**

**i. What is the use of format specifiers? give examples.**

Format specifiers are used as place holders for different types of variables. They are used in output functions for concatenation of variables values with the string and in input functions for picking right variable value.

Examples:

%d or %i	Signed integer	%ld	Long integer
%f	Float(decimal value)	%g or %e	Exponential
%c	Single character	%s	String

**ii. Why escape sequence are used? Give examples.**

**Escape Sequence** are letters that start with a backslash ( \ ). They control the print output on the screen and itself do not get printed. They are used with strings enclosed with in double quotes.

**Examples:** \n for new line, \t for tab (space) , \a for alert beep, \b for one backspace etc.

**iii. What is the purpose of gets() function?**

gets() function is used to get a string input. It is terminated as the user presses the enter key.

Its syntax is: **gets(character\_array\_name);**

**iv. Differentiate between getch() and getche() functions.**

**getch() and getche()** both reads a single character from the keyboard. However, getche() displays the character on the screen while getch() **does not** display it on the screen.

**v. Evaluate the following expressions.**

a.  $9-5*(6+2) = 9-5*8 = -31$

b.  $60/10*24+3 = 6*24+3 = 144+3 = 147$

c.  $100\%50-100\%3 = 0- 1 = -1$

**vi. Differentiate between simple and compound assignment operator?**

**Simple assignment operator:**

The simple assignment operator ( = ) is used to assign a value to a variable. It can assign int, float or character value but cannot assign string value.

**Its syntax is :** **variable = value( int , float or char).**

**Compound assignment Operator:**

Compound assignment operator combines the assignment operator with any arithmetic operator like + , - , \* , /. It is used to perform mathematical operations easily.

e.g. **a+=10;** (same as a=a+10). **a-=10;** (same as a=a-10). **a\*=10;** (same as a=a\*10).

**Q.3 what are operators? Explain different types of operators with examples.**

**Ans:** operators are symbols that are used to perform certain operations on data. C language has the following types of operators: arithmetic , assignment , relational and logical.

#### Arithmetic operators:

Arithmetic operators perform mathematical operations on data(operands).they work with numeric values. The arithmetic operators in C are listed below:

Operation	Symbol	Description	example
Addition	+	Add numbers	a+b
Subtraction	-	Subtract one number form another	a-b
Multiplication	*	Multiply two numbers	a*b
Division	/	Divides one value by another	a/b
Modulus /Remainder	%	Finds remainder when one value is divided over another.	a%3

#### Assignment Operator:

The assignment operator (=) is used to assign a value to a variable. It can assign int, float or character value but cannot assign string value.

**Its syntax is : variable = value( int , float or char).**

#### Compound assignment Operator:

Compound assignment operator combines the assignment operator with any arithmetic operator like + , - , \* , /. It is used to perform mathematical operations easily.

e.g. **a+=10;** (same as a=a+10). **a-=10;** (same as a=a-10). **a\*=10;** (same as a=a\*10).

#### Increment Operator:

The increment operator is represented by double plus ++ sign. It adds one to the variable value.

x++ equivalent to x= x+1;

#### Decrement Operator:

The decrement operator is represented by a double – sign. It subtracts 1 from the variable value.

x- - is equivalent to x =x-1;

#### Relational Operators:

Relational operators are used to specify conditions in program by comparing two values. They are also called conditional operators. They return a TRUE or FALSE result. Some relational operators are below:

> greater than, < less than, == equals, >= greater than or equal. <= less than or equal, != not equal.

#### Logical Operators:

Logical operators evaluate compound conditions. There are three logical operators. AND , OR ,NOT.

**AND** operator returns TRUE if both conditions are true, otherwise it returns FALSE.

**OR** operator returns TRUE if any of the condition is true. If both conditions are false, it will return false.

**NOT** operator returns TRUE if condition is false, and FALSE if condition is true.

#### Q.4 What is output function? Describe in detail different types of output functions.

**Output** functions are used to write data into computer screen or to display messages to the user. C language has following output functions:

- Printf()
- Puts()
- cout
- putchar()

**Printf()** function is a common output function. It can display variable values and messages on the screen. Its syntax is ***printf("string message", variables);***

**Puts()** function is used to print string on the screen. Its syntax is ***puts(string);***

The string may be constant or variable. Constant will be enclosed in double quotes.

**Cout** is an object in C++ defined by ***iostream.h*** header file. It is used to print anything like variables or constants on the screen. Its syntax is : ***cout<<variable or constant or "string";***

**putchar() function** is used to print a single character on the screen. e.g. ***putchar('a');***

#### Q.5 Discuss different format specifiers and escape sequences used in printf () function.

**Format specifier** is a conversion specification code. It is used with input and output functions to tell what data type it will expect. Some commonly used format specifiers are below:

- %d or %i for signed integer
- %ld for long integer
- %f for floating point(decimal)
- %e for exponential notation
- %c for single character
- %s for string

**Escape Sequence** are letters that start with a backslash( \ ). They control the print output on the screen and itself do not get printed. They are used with strings enclosed with in double quotes. commonly used escape characters are as below:

- \n for new line
- \t for tab (space)
- \a for alert beep
- \b for one backspace etc.
- \' for single quote
- \" for double quote.
- \\ for backslash

#### Q.6 Explain different types of input functions with examples.

Input functions are used to get value from the user during program execution and assign it to variables. The input function used in C are listed below:

- Scanf()

- getch()
- getche()
- getchar()
- gets()
- cin

**scanf() function:** scanf() is an input function that accepts values during program execution. It can take different values specified by format specifiers. scanf() pauses the execution of the program until a value is entered by the user. this function does not accept space.

Syntax of scanf() function is: `scanf("format specifiers",&var1,&var2,...);`

e.g. `scanf("%d,%d",&a,&b);`

**getch() , getche() and getchar() functions:**

all these functions take single character from keyboard. getchar() waits for Enter Key press while getch() and getche() do not wait for Enter key press. Getch() do not echo(display) the character on the screen while getche() display the character on the screen.

E.g. `c = getche();` , `c = getch();` `c = getchar();`

**gets() function:**

this function reads an entire string from keyboard until the Enter key is pressed by the user. It can accept spaces and tabs.

Syntax : `gets(string_array);`

**Cin :** is used in c++ to obtain input form keyboard runtime. Cin can be used to obtain any kind of value. it does not need any format specifiers. *iostream.h* header file is necessary for cin to work.

Syntax: `cin>>var1>>var2>>var3...;`

**Q.7 Write a program that reads three numbers and prints their sum , product and average.**

```
#include<stdio.h>
int main(){
    int a,b,c,sum,prod,avg;
    printf("Enter first Number:");
    scanf("%d",&a);
    printf("\nEnter 2nd Number:");
    scanf("%d",&b);
    printf("\nEnter 3rd Number:");
    scanf("%d",&c);
    sum = a+b+c;
    prod = a*b*c;
    avg = sum/3;
    printf("\n The sum is: %d",sum);
    printf("\n The product is: %d",prod);
    printf("\n The average is: %d",avg);
}
```

Q.8 Write a program that reads the base and height of a triangle and prints its area.

```
#include<stdio.h>
int main(){
    int b,h,area;
    printf("Enter base of the triangle:");
    scanf("%d",&b);
    printf("\nEnter height of the triangle:");
    scanf("%d",&h);
    area = b*h/2;
    printf("\n The area is: %d",area);
}
```

Q.9 Write a program that reads temperature in Celsius, converts it into Fahrenheit and prints on the screen.

```
#include<stdio.h>
int main(){
    float c,f;
    printf("Enter the temprature in celsius:");
    scanf("%f",&c);
    f= (c*9/5)+32;
    printf("\n The Temprature in Fahrenhiet is: %.2f",f);
}
```

Q.10 Write a program that reads name and address of a student and prints it on the screen using gets() and puts() functions.

```
#include<stdio.h>
int main(){
    char name[15];
    char add[20];
    printf("Enter name of the student:");
    gets(name);
    printf("\nEnter Address of the student:");
    gets(add);

    printf("\n Name of the student is: %s",name);
    printf("\n Address of the student is: %s",add);
}
```

Notes Prepared by: IQBAL AHMAD (SST-IT) /BS - Computer Science  
Note: Errors, Improvements, suggestions and Feedback about the notes will be welcomed at:  
[iqbalahmadq@gmail.com](mailto:iqbalahmadq@gmail.com)

Q2: Write a program in C to input three integers. Find out the largest among these integers using if-else structure and print it on the screen.

```
#include"stdio.h"
int main(){
    int x,y,z,l;
    printf("Enter three numbers:");
    scanf("%d %d %d",&x,&y,&z);
    if(x>y)
        l= x;
    else
        l= y;
    if(l>z)
        printf("the largest number is:%d",l);
    else
        printf("the largest number is:%d",z);
}
```

Q.3 write a program in C to input a single character and print a message "it is vowel" or "it is a consonant" using if else structure.

```
#include"stdio.h"
#include"conio.h"
int main(){
    char ch;
    printf("Enter any aplhabet (a-z): ");
    ch = getch();
    if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u')
        printf("\n You have entered a vowel");
    else
        printf("\n You have entered a consonant");
}
```



Q.4 Write the program in Q3 using switch-case structure.

```
#include"stdio.h"
#include"conio.h"
int main(){
    char ch;
    printf("Enter any aplhabet (a-z): ");
    ch = getche();
    switch(ch){
        case 'a':
        case 'e':
        case 'i':
        case 'o':
        case 'u':
            printf("\n You have entered a vowel");
            break;
        default:
            printf("\n You have entered a consonant");
    }
}
```

Q.5 write the output of the following program.

```
#include"stdio.h"
#include"conio.h"
int main(){
    int a,b,c;
    a=b=10;
    c=13;
    if((a==b)&&(a>c))
        printf("condition is true\n");
    else
    {
        printf("a is equal to b \n");
        printf("a is not greater than c \n");
    }
}
```

Out put:

a is equal to b

a is not greater than c

Q.6 Find out Errors if any in the following program:

```
#include "stdio.h"
#include "conio.h"
int main(){
int a;
printf("enter any number=");
scanf("%f",&a);
if(a>10)
printf("value is greater than 10");
else if
printf("value is greater than 10");
}
```

Ans : Error(1) : on line number 6, the format specifier for int is wrong. It should be "%d".

Error(2) : on line number 9, the "if" after else is unnecessary, this should be removed;

Notes Prepared by: IQBAL AHMAD (SST-IT) [BS - Computer Science]

Note: Errors, Improvements, suggestions and Feedback about the notes will be welcomed at:

[iqbalahmaddg@gmail.com](mailto:iqbalahmaddg@gmail.com)

**Q2: Explain different types of looping structure with examples.**

**Loop:** The type of control structure that repeats a statement or set of statements is known as loop.

During programming, it sometimes happens that we need to repeat one or more statements for a specified number of times and loop structure is used here for this repetition purpose.

Loops are basically used in two ways; to repeat for some specific number of times. Or to repeat until a certain condition based on TRUE or FALSE is matched.

Three types of loops are used in C. *for loop, while loop, Do-while loop.*

**The “for” loop:**

The for loop executes one or more statements for a specified number of times. its syntax is :

**for(initialization;condition;inrement/decrement)**

{

**Body of loop(Statements)**

}

Explanation:

**Initialization** is an expression to set initial value of the counting variable.

**Condition** is a relational expression, if it is true loop will run otherwise loop will stop.

**Increment/Decrement** changes the value of counter after each iteration.

**Body of loop** is the set of statements enclosed in braces { } that to be repeated.

Example program: showing numbers from 1 to 10.

```
#include <stdio.h>
void main(){
int x;
    for(x=1; x<=10; x++){
        printf("%d \t",x);
    }
}
```

**The while loop:**

The is useful when number of iterations are not known in advance. It executes until a certain condition remains true. Its syntax is as following:

**While(condition){**

**Body of loop(statements)**

}

Explanation:

**Condition:** it is a Boolean expression, the body of the loop will be executed only if this condition is true. Otherwise it will stop executing.

**Body of loop** is the set of statements enclosed in braces { } that to be repeated.

Example program that display "Pakistan" five times.

```
#include <stdio.h>
void main(){
int n = 1;
    while(n<=5){
        printf("Pakistan \n");
        n++;
    }
}
```

### The do-while loop:

In do-while loop the condition comes after the body, therefore it will run at least once. The rest of its function is same as while loop. Its syntax is as follows:

```
do {
    Body of loop(statements)
} while(condition)
```

#### **Explanation:**

**do** is a keyword that indicates the beginning of the loop.

**Condition:** it is a Boolean expression; the body of the loop will keep executing only if this condition is true. Otherwise it will execute only once.

**Body of loop** is the set of statements enclosed in braces { } that to be repeated.

Example program that print first ten integers using do while loop.

```
#include <stdio.h>
void main(){
int x = 1;
    do{
        printf("%d \n",x);
        x++;
    } while(x<=10)
}
```

Q.3 differentiate between the *do-while* and *while* loop. Explain with program examples

**Ans:** The main difference between *while* and *do-while* loop is that, *while* loop will run if the condition is true otherwise it will not execute. While the *do-while* loop will run at least once in all cases. Even if the condition is initially false, it will make one iteration.

Example programs:

Example program that display "Pakistan" five times using while loop.

```
//WHILE LOOP
#include <stdio.h>
void main(){
    int n = 1;
    while(n<=5){
        printf("Pakistan \n");
        n++;
    }
}
```

Example program that print first ten integers using do while loop.

```
//DO-WHILE LOOP
#include <stdio.h>
void main(){
    int x = 1;
    do{
        printf("%d \n",x);
        x++;
    } while(x<=10)
}
```

Q.4 Write a program in C to calculate and print the product of the even numbers from 1 to 100 by using *while* loop.

```
#include"stdio.h"

int main(){
    int c;
    double product=1;
    c=2;
    while (c<=100)
    {
        product=product*c;
        c=c+2;
    }
    printf("product = %e",product);
}
```

Q.5 Write a program to display alphabets from A to Z using for loop.

```
#include"stdio.h"
int main(){

    char alpha;
    for(alpha= 'A';alpha<='Z';alpha++)
    printf("%c\t",alpha);
}
```

Q.6 write a program to display the following output using nested for loop.

```
a
a  a
a  a  a
a  a  a  a
a  a  a  a  a
```

```
#include<stdio.h>
int main(){
    int i,j;
    for(i = 1; i <= 5; i++){
        for(j = 1; j <= i; j++)
            printf("a \t");
        printf("\n");
    }
}
```

Q.7 Write a program in C to print the sum of odd numbers from 1 to 50 using for loop.

```
#include"stdio.h"

int main(){
    int sum =0;

    for(int i=1; i <=50; i+=2)
    {
        sum+=i;
    }
    printf("Sum = %d",sum);
}
```

Notes Prepared by: IQBAL AHMAD (SST-IT) [BS - Computer Science]  
 Note: Errors, Improvements, suggestions and Feedback about the  
 notes will be welcomed at:  
[iqbalahmadda@gmail.com](mailto:iqbalahmadda@gmail.com)

**Q2. How is data represented in a computer, briefly explain?**

**Ans:** The raw facts and figures processed by digital computer is called data. Before processing and saving data in digital computers, it is first converted into binary digit system i.e. 0s and 1s. It is then represented inside digital computers as **ON** and **OFF** electrical pulses. **ON** represents **1** while **OFF** represents **0**.

One digit in binary number system is called a **bit** and a combination of 8 bit is called a **Byte**. Byte is the basic unit for measuring computer memory. One byte can hold one character of data for example an alphabet, numeric or any special symbol etc.

If we have to save the letter A in computer memory, it will be converted to its binary specific code i.e 01000001 and then saved in memory. And same is the case for all.

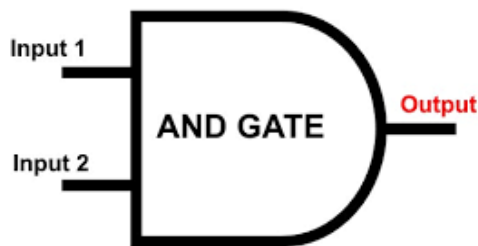
**Q3. What are the three basic gates draw their circuit diagrams and truth tables.**

**Logic gate:** An Electronic circuit that takes one or more input signals and produce single output is called a logic gate.

The three Basic Logic gates are: **AND , OR , NOT**. They are basic gates since they can be combined to create any other logic circuit.

**AND GATE:**

The AND gate performs logical Multiplication. It gives us output 1 when all its inputs are 1. If any input becomes 0, output will be zero. Its function is represented as  $F = X.Y$  ( . ) means logical multiplication.



A	B	Z
0	0	0
0	1	0
1	0	0
1	1	1

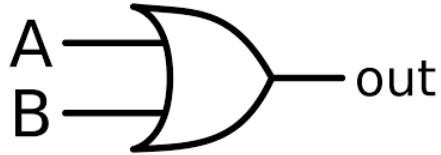
(2 input AND Gate truth Table)

A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

(3 input AND Gate truth Table)

**OR GATE:**

OR gate performs logical addition. It gives 1 output If any of the input is 1. If all input becomes zero, it will produce 0 output. Its function is represented by  $F = X+Y$  (where + symbol represents logical addition).



A	B	Z
0	0	0
0	1	1
1	0	1
1	1	1

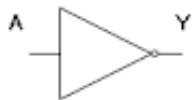
2 inputs OR gate Truth Table

A	B	C	Z
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

3 inputs OR gate Truth Table

**NOT GATE:**

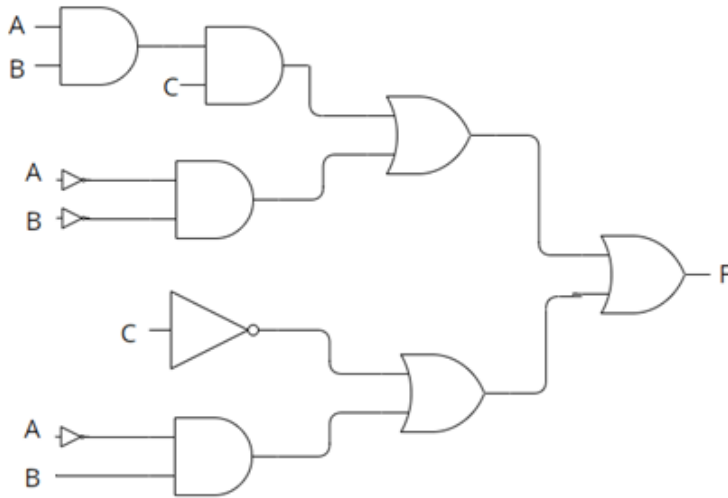
NOT gate performs the logical Negation. It reverses the input. If its input is 1 output will be 0 and if input is 0 output will be 1. In other words, it gives us complement of the input.



Truth Table	
A	A'
1	0
0	1

**Q4. Convert the following Boolean expression to logic gate.  $f(A,B,C) = ABC + \bar{A} \bar{B} + \bar{C} + \bar{A} B$**





Q5. Simplify the following Boolean functions using K-Map method. Also, draw the logic diagram of the simplified function.

i.  $F = \bar{X}Z + \bar{X}\bar{Z}$

X	Z	F
0	0	1
0	1	1
1	0	0
1	1	0

X \ Z	0	1
0	1	0
1	1	0

Simplified Function From K-Map:  $F(X,Y,Z) = X'$



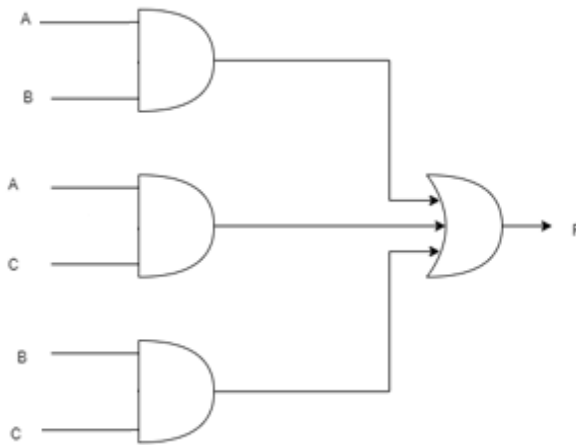
SIMPLIFIED LOGIC DIAGRAM

ii.  $F = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

		AB			
		00	01	11	10
C	0	0	0	1	0
	1	0	1	1	1

Simplified Function From K-Map:  $F(A,B,C) = AB+AC+BC$



**Q2. What is website. Describe different types of websites with examples.**

**Ans:** a website is a collection of webpages hosted on a webserver. Each website has an address associated with it called URL which is unique for every website. Any website on the internet can be accessed with its URL.

**Types of Websites:**

**Web portals:** web portals are websites that offer different services like online shopping, University Learning Management systems to deliver information about courses and other progress to the students.

Example: Sarhad University Peshawar LMS <https://delms.suit.edu.pk/>

**News websites:** News websites present published news to the visitors. Readers are also allowed to publish their stories and other news. Example: <https://dailymashriq.epapers.pk/>

**Informational Websites:** those websites, which provide information almost on every topic in the form of text, graphics or videos. Example: <https://www.wikipedia.org/>

**Educational Websites:** Educational websites contain animation ,slide presentations and tutorials on various topics to educate people. Example: <https://www.ilmkidunya.com/>

**Business websites:** Business websites provide facilities to maintain business relationships and selling information, services and commodities via internet. These sites are used to exchanges business documents like orders and sale invoices etc.

**Entertainment Websites:** These website provide different recreational facilities like videos and online games etc. example. [www.youtube.com](http://www.youtube.com)

**Q3. Write html tags for the following:**

- i. **Paragraph**  
<p></p>
- ii. **Heading**  
<h1></h1>, <h2></h2> .... <h6></h6>
- iii. **Bold**  
<b></b>
- iv. **Underline**  
<u></u>
- v. **Italic**  
<i></i>
- vi. **Center text**  
< h1 align="center"></h1>
- vii. **Superscript**  
<sup></sup>
- viii. **Subscript**  
<sub></sub>
- ix. **Font size , color , typeface**  
<font size = 3></font>

```
<font color = red></font>
```

```
<font face = arial></font>
```

**Q4. Create a webpage I html that displays image of printer. width of image should be 250 pixels and height 150 pixels.**

**HTML Script(code) :**

```
<html>
<body>
<h1>Showing printer image</h1>

</body>
</html>
```

**Q5. describe how background color and image are applied to a webpage?**

**Back ground color:** the background color of a page can be set using *bgcolor* attribute on the <body> tag.

The small image is repeated in the browser window to fill it.

**Example :** <body bgcolor = red text = green>

**Background Image:** The background of a page is set using “background” attribute on the <body> tag.

**Example :** <body background = “image.png”>

**Q6. Create an HTML document that contains a graphical hyperlink.**

```
<html>
<body>
<h2>Graphical Hyperlink</h2>
<p>Using image as a link</p>
<a href="https://bisess.edu.pk"></a>
</body>
</html>
```

**Q7. Create a webpage in HTML that contains unordered and ordered lists**

```
<html>
<body>
<h2>An Unordered HTML List</h2>
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
<h2>An Ordered HTML List</h2>
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
</body>
</html>
```

**Q8. Create a webpage in HTML that contains hyperlink to three websites.**

```
<html>
<body>
<h1>HTML hyper Links</h1>
<p><a href="https://www.google.com/">Google</a></p>
<p><a href="https://www.facebook.com/">Facebook</a></p>
<p><a href="https://kpese.gov.pk/">KP ESED</a></p>
</body>
</html>
```

**Q9. Create a webpage in HTML to display your class timetable.**

```

<html>
<head>
<title>table in html</title>
<body>

<table border=1 cellspacing=7 cellpadding=8>
<tr style ="height:70" align=center>
<td colspan="9" valign=center> <b> <i>Time table class 10th GHS SAWAWAI BUNER session 2021-
22</i> </b> </td>
</tr>

    <tr>
        <td>day/period</td>
        <td align=center width= 100>1</td>
        <td align=center width= 100>2</td>
        <td align=center width= 100>3</td>
        <td align=center width= 100>4</td>
        <td align=center width= 100>5</td>
        <td align=center width= 100>6</td>
        <td align=center width= 100>7</td>
        <td align=center width= 100>8</td>
    </tr>
    <tr align=center>
        <td>Monday</td>
        <td>english</td>
        <td>islamiat</td>
        <td>cs</td>
        <td>urdo</td>
        <td>maths</td>
        <td align=center>ps</td>
        <td>chemistry</td>
        <td>physics</td>
    </tr>

```

```
<tr align=center>
  <td>Teusday</td>
  <td>english</td>
  <td>islamiat</td>
  <td>cs</td>
  <td>urdo</td>
  <td>maths</td>
  <td align=center>ps</td>
  <td>chemistry</td>
  <td>physics</td>
</tr>
<tr align=center>
  <td>wednesday</td>
  <td>english</td>
  <td>islamiat</td>
  <td>cs</td>
  <td>urdo</td>
  <td>maths</td>
  <td align=center>ps</td>
  <td>chemistry</td>
  <td>physics</td>
</tr>
<tr align=center>
  <td>Thursday</td>
  <td>english</td>
  <td>islamiat</td>
  <td>cs</td>
  <td>urdo</td>
  <td>maths</td>
  <td align=center>ps</td>
  <td>chemistry</td>
  <td>physics</td>
</tr>
```

```
<tr align=center>
    <td>Friday</td>
    <td>english</td>
    <td>chemistry</td>
    <td>cs</td>
    <td>urdo</td>
    <td>maths</td>
<td>--</td>
<td>--</td>
<td>--</td>

</tr>

<tr align=center>
    <td>Saturday</td>
    <td>english</td>
    <td>ps</td>
    <td>cs</td>
    <td>urdo</td>
    <td>maths</td>
    <td align=center>ps</td>
    <td>chemistry</td>
    <td>physics</td>

</tr>

</body>
</head>
</html>
```

Notes Prepared by: IQBAL AHMAD (SST-IT) [BS - Computer Science]

(software Developer; Android ,Desktop and web Apps)

Note: Errors, Improvements, suggestions and Feedback about the notes will be welcomed at: [iqbalahmaddg@gmail.com](mailto:iqbalahmaddg@gmail.com)