CH # 1 INTRODUCTION TO PROGRAMMING

Topic No.		Page No.
1.1	**PROGRAMMING ENVIRONMENT** 1.1.1 Integrated Development Environment 1.1.2 Text Editor 1.1.3 Compiler	2
1.2	**PROGRAMMING BASIC** 1.2.1 Reserved Word 1.2.2 Structure of a C program 1.2.3 Purpose and Syntax of comments in C Program	7
1.3	**CONSTANTS AND VARIABLES** 1.3.1 Constants 1.3.2 Variables 1.3.3 Data types of a variable 1.3.4 Name of a variable 1.3.5 Variable Declaration 1.3.6 Variable Initialization	14
*	**EXERCISE**	23
*	**PROGRAMMING EXERCISE**	28

www.ilmkidunya.com

1.1 PROGRAMMING ENVIRONMENT

LONG QUESTIONS

1. Explain Programming Environment in detail. (K.B+U.B)

Ans: A collection of all the necessary tools for programming makes up a programming environment. It is essential to setup a programming environment before we start writing programs. It works as a basic platform for us to write and execute programs.

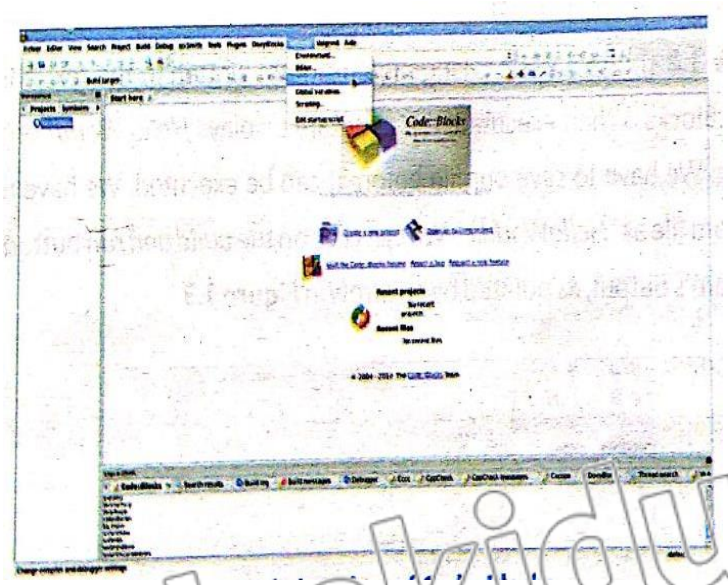
Integrated Development Environment (IDE):

A software that provides a programming environment to facilitate programmers in writing and executing computer programs is known as an Integrated Development Environment (**IDE**).

An **IDE** has a graphical user interface (**GUI**), meaning that a user can interact with it using windows and buttons to provide input and get output. An **IDE** consists of tools that help a programmer throughout the phases of writing, executing and testing a computer program. This is achieved by combining text editors, compilers and debuggers in a single interface.

Some types of the many available **IDEs** for C programming language are:

1. Visual Studio
2. Xcode
3. Code::Blocks
4. Dev C + +

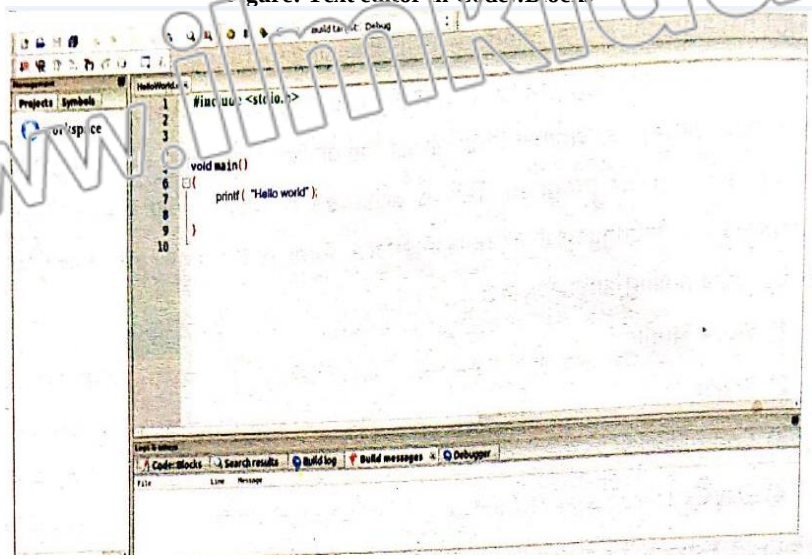


Text Editor:

A text editor is a software that allows programmers to write and edit computer programs. All IDEs have their own specific text editors. It is the main screen of an IDE where we can write our programs.

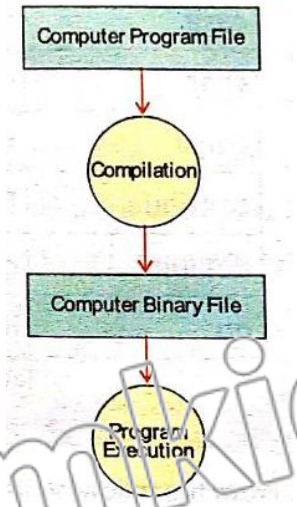
www.ilmkidunya.com

Figure: Text editor in Code::Blocks



Compiler:

Computers only understand and work in machine language consisting of 0's and 1's. They require the conversion of a program written in programming language to machine language, in order to execute it. This is achieved using a compiler. A compiler is a software that is responsible for conversion of a computer program written in some high level programming language to machine language code.



www.ilmkidunya.com

SHORT QUESTIONS

Q.1 Define computer program or computer software. (K.B)

Ans: Computers need to be fed a series of instructions by humans which tell them how to perform a particular task. These series of instructions are known as a computer program or software.

Q.2 What is a programmer? (K.B)

Ans: The person who knows the detail about the syntax of the programming language and can write a program is called a programmer.

Q.3 What are programming languages? (K.B)

Ans: Computer programs are written in languages called programming languages. Some commonly known programming languages are

- Java
- C
- C+ +
- Python.

Q.4 Name any five programming language. (K.B)

Ans:

- C language
- C++
- C#
- JAVA
- Python

Q.5 Define programming Environment. (K.B)

Ans: A collection of all the necessary tools for programming makes up a programming environment. It is essential to setup a programming environment before we start writing programs. It works as a basic platform for us to write and execute programs.

Q.6 Define IDE. (K.B)

Ans: A software that provides a programming environment which facilitates the programmer in writing and executing computer programs is known as an Integrated Development Environment (IDE).

Q.7 Name some commonly available IDE of C. (K.B)

Ans:

- visual studio
- code:: blocks
- x code
- Dev C ++
- Turboo C

Q.8 Define Text editor. (K.B)

Ans: A text editor is a software that allows programmers to write and edit computer programs. All IDEs have their own specific editors.

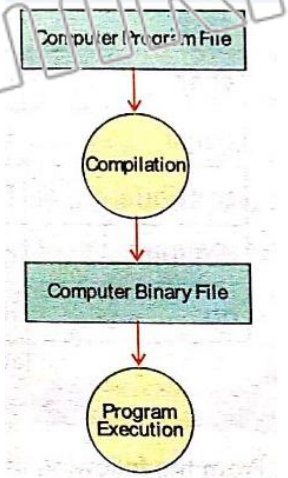
Q.9 Define compiler. (K.B)

Ans: A compiler is a software that is responsible for conversion of a computer program written in some programming language to machine language code.

www.ilmkidunya.com

Q.10 Show program execution with help of flowchart diagram. (K.B+U.B)

Ans:



Q.11 How many modules do 'C' IDE has? (K.B+U.B)

Ans: C IDE has 5 modules.

- Text Editor
- Compiler
- Linker
- Loader
- Debugger

MULTIPLE CHOICE QUESTIONS

1. The process of writing a computer program in computer programming language to solve a particular problem is called _____. (K.B)

(A) Programming (B) Software Testing
(C) Software Maintenance (D) Software Documentation

2. Which of following is related to computer program? (K.B+U.B)

(A) Solve Particular Problem (B) Developed in Computer language
(C) Coding (D) All of Above

3. The computer languages that are used for writing program are called _____. (K.B)

(A) Natural Language (B) Programming Language
(C) Native Language (D) National Language

4. A set of instruction writes in programming language is called _____. (K.B)

(A) Program (B) Software Design (C) Algorithm (D) Flowchart

5. Each statement of programming language has its own _____. (K.B)

(A) Syntax (B) Semantic (C) Coding (D) Both 'A' & 'B'

6. Computer microprocessor execute to solve a particular problem _____. (K.B+U.B)

(A) Set of Instruction (B) Data (C) Algorithm (D) Flowchart

7. The rules of programming language according to which statement of a program is written, called as _____. (K.B)

(A) Syntax (B) Format (C) Semantic (D) Algorithm

www.ilmkidunya.com

8. The grammatical rules of a natural language are similar to programming language. (K.B)
(A) Algorithm (B) Pseudo Code (C) Documentation (D) Syntax

9. A language that is understandable by computer is called: (K.B)
(A) Native Language (B) National
(C) Programming Language (D) Local Language

10. The use of computer programming language includes: (K.B+U.B)
(A) Write Program
(B) Use for Communication between User and Computer
(C) Use to Control System
(D) All of Above

11. The computer programming languages are classified into categories: (K.B)
(A) Two (B) Three (C) Four (D) Five

12. Some of the variable IDEs of 'C' are. (K.B)
(A) Visual Studio (B) X code (C) code block (D) All

13. A set of processes and programming tools used to develop computer program is called. (K.B+U.B)
(A) Programming Environment (B) Programming Skills
(C) Programming Tools (D) Programming Techniques

14. To create, compile and run a program in programming language we use. (K.B)
(A) IDE (B) Standard Tools (C) GUI (D) Text Editor

15. In C language IDE consist of the modules: (K.B)
(A) Two (B) Four (C) Three (D) Five

16. A simple word processor that is used to create and edit source code of a program. (K.B)
(A) Text Editor (B) Word Processor (C) Microsoft Word (D) Spread Sheet

17. A computer software that translates 'C' language program into machine code. (K.B)
(A) Compiler (B) Interpreter (C) Linker (D) Assembler

18. Which one of the following is not an IDE of 'C'? (K.B)
(A) Visual studio (B) X code (C) Dev C # (D) C #

19. GUI stands for (K.B+U.B)
(A) Graphical Use Internet (B) Good Using Interface
(C) Graphical User Interface (D) Graphical Using Internet

20. IDE allows a user to interact with it using windows and button for inputs and outputs. (K.B)
(A) GUI (B) CLI (C) MDI (D) All of these

21. provides an interface to a user. (K.B)
(A) IDE (B) Text Editor (C) Compiler (D) None of these

22. IDE is a combination of. (K.B+U.B)
(A) Text Editors (B) Compiler (C) Debuggers (D) All of these

23. Some commonly used programming languages are? (K.B)
(A) C (B) C++ (C) C# (D) All of these

24. C is a _____ language. (K.B)
(A) Basic (B) High level (C) Programming (D) Both B and C

25. allows a programmer to write and edit a program. (K.B)
(A) Text editor (B) compiler (C) IDE (D) All of these

26. All IDE's have _____ text Editors. (K.B)
(A) Same (B) Unique (C) Both A & B (D) None of these

27. is the main screen of IDE. (K.B)
(A) Text Editor (B) Compiler (C) Debugged (D) None of these

www.ilmkidunya.com

28. Computers only understands _____. **(K.B)**
(A) 0 and 1 (B) English (C) Roman language (D) None of these

29. _____ converts a programming language to machine code. **(K.B)**
(A) Compiler (B) Text Editor (C) Debugger (D) All of these

30. C language uses a _____. **(K.B+U.B)**
(A) Compiler (B) Interpreter (C) Both A & B (D) None of these

31. Program written in high level language is called _____. **(K.B)**
(A) Source code (B) Object code (C) Both A & B (D) None of these

32. Program written in low level language is called _____. **(K.B)**
(A) Source code (B) Object code (C) Both A & B (D) None of these

1.2 PROGRAMMING BASIC

SHORT QUESTIONS

Q.1 Define syntax. **(K.B)**

Ans: Every programming language has some primitive building blocks and follows some grammar rules known as its syntax.

Q.2 Define syntax error with the help of example. **(K.B)**

Ans: While programming, if proper syntax or rules of the programming language are not followed, the program does not get compiled. In this case, the compiler generates an error. This kind of errors are called syntax errors.

Example:
Typing **'pintf'** instead of **'printf'**

MULTIPLE CHOICE QUESTIONS

1. ____ is the set of rules to write a program. **(K.B)**
(A) syntax (B) semantic (C) X code (D) None of these

2. Program does not gets compiled in case of _____ errors. **(K.B)**
(A) Syntax (B) Logical (C) Both (D) None of these

3. 5=a is a type of _____ error. **(K.B+U.B)**
(A) Syntax (B) Logical (C) Routine (D) None of these

4. Syntax errors are also known as ____ errors. **(K.B)**
(A) Typographical (B) Exaction (C) Silly (D) Del

1.2.1 RESERVED WORDS

SHORT QUESTIONS

Q.1 Define reserved words. **(K.B)**

Ans: Every programming language has a list of words that are predefined. Each word has its specific meaning already known to the compiler. These words are known as reserved words or keywords.

Q.2 How many keywords a C language have? **(K.B)**

Ans: C language has 32 reserved words

Q.3 Name any five reserved words **(K.B)**

Ans:

- if
- int
- auto
- break
- do

www.ilmkidunya.com

Q.4 Write all 32 reserved words of C. **(K.B)**

Ans:

auto	Double	int	struct
break	Else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

MULTIPLE CHOICE QUESITONS

1. In 'C' language word that are part of programming language and have special use: (K.B+U.B)

(A) Special Character (B) Digits

(C) Constants (D) Reserve Words

2. The number of reserved words used in 'C' language are _____. **(K.B)**

(A) 20 (B) 32 (C) 31 (D) 42

3. Which one of the following is not a reserve word of 'C' language? **(K.B)**

(A) auto (B) char (C) int (D) x

4. Reserved words are also called _____. **(K.B)**

(A) Keyboards (B) Predefined (C) Both 'A' & 'B' (D) None of these

1.2.2 STRUCTURE OF C PROGRAM

LONG QUESTIONS

1. Explain structure of 'C' program. **(K.B+U.B)**

Ans:

STRUCTURE OF A 'C' PROGRAM

A 'C' program is divided into three main parts:

1. **Link section or header section:** While writing programs in 'C' language, we make extensive use of functions that are already defined in the language. But before using the existing functions, we need to include the files where these functions have been defined. These files are called header files. We include these header files in our program by writing the include statements at the top of program.
   General structure of an include statement is as follows:
   #include<header_file_name>
2. **Main section:** It consists of a **main ()** function. Every 'C' program must contain a main () function and it is the starting point of execution.
3. **Body of main () function:**
   The body of main () is enclosed in the curly braces {}. All the statements inside the curly braces make the body of main function.
   Example:
   # include <stdio.h>
   void main ( )

www.ilmkidunya.com

```
{
printf ("Hello World!");
}
```

In the above program the statement **printf ("Hello World!");** uses a predefined function printf to display the statement **Hello World!** on computer screen. We can also create other functions in our program and use them inside the body of **main ()** function.

SHORT QUESTIONS

Q.1 How many parts do a C program have? (K.B)

Ans: C program has three parts

1. Link section or header section
2. Main section
3. Body of main ( ) function

Q.2 Define Link section or header section. (K.B)

Ans: While writing programs in 'C' language, we make extensive use of functions that are already defined in the language. But before using the existing functions, we need to include the files where these functions have been defined. These files are called header files. We include these header files in our program by writing the include statements at the top of program.

General structure of an include statement is as follows:

#include<header_file_name>

Example:

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

Q.3 Define Main section of C program. (K.B)

Ans: It consists of a main () function. Every 'C' program must contain a main () function and it is the starting point of execution.

Example:

```
void main ( )
```

Q.4 What is body of Main () function? (K.B)

Ans: The body of **main ()** is enclosed in the curly braces **{}**. All the statements inside the curly braces make the body of main function.

Example:

```
{
printf ("Hello World!");
}
```

In the above program, the statement *printf* ("Hello World!"); uses a predefined function *printf* to display the statement Hello World! on computer screen.

Q.5 What points should be kept in mind while developing a C program? (K.B+U.B)

Ans: Following points must be kept in mind in order to write syntactically correct C language programs.

- The sequence of statements in a C language program should be according to the sequence in which we want our program to be executed.
- C language is case sensitive. It means that if a keyword is defined with all small case letters, we cannot capitalize any letter i.e. int is different from int. Former is a keyword, whereas latter is not.
- Each statement ends with a **semi-colon(;)** symbol.

www.ilmkidunya.com

Q.6 Identify different parts of the following C program. (U.B)

Ans:

Program	Parts of C program
#include <stdio.h>	//pre processor directives
#include <conio.h>	//preprocessor directives
void main ( )	//main function
{	//body of fiction {} are known as delimiters
printf("I am a student of class 10");	"); //printfunction // ("") referred as string literals
getch ( );	// input function
}	

MULTIPLE CHOICE QUESITON

1. How many types of header files are there in 'C' language program? (K.B)
(A) 2 (B) 4 (C) 3 (D) Many

2. "math.h" header file contains function: (K.B)
(A) printf ( ) (b) getch ( ) (c) clrscr ( ) (d) sin ( )

3. Which one of the following is included in "conio.h" header file of 'C'? (K.B)
(A) getche ( ) (b) scanf ( ) (c) pow ( ) (d) log ( )

4. Which one of the following is not a part of 'C' program? (K.B+U.B)
(A) Preprocessor Directives (B) Body of Program
(C) Main ( ) Function (D) Compiler

5. In 'C' program the instructions for compiler is called: (K.B)
(A) Preprocessor Directive (B) Body of main ( )
(C) Main ( ) Function (D) Syntax

6. In 'C' language preprocessor directive starts with a ____. (K.B)
(A) ; (B) # (C) : (D) !

7. The most commonly used preprocessor directive includes: (K.B)
(A) Includes (B) Header File (C) Define (D) Both 'A' & 'C'

8. In 'C' program the body of main function surrounded by: (K.B)
(A) ( ) (B) { } (C) [ ] (D) !

9. There are _____ sectors of a 'C' program. (K.B)
(A) 1 (B) 2 (C) 3 (D) 4

10. "stdio.h" consists of basic _____ functions. (K.B)
(A) Input (B) Output (C) Both 'A' & 'B' (D) None of these

11. _____ is enclosed in curly braces. (K.B)
(A) Body of main (B) Main (C) Link section (D) None of these

12. Syntax of header file. (K.B)
(A) #include <header – file –name> (B) # include <header >
(C) <header-file-name> (D) None of these

13. Starting point of execution is _____. (K.B+U.B)
(A) Main () function (B) header file (C) link (D) None of these

14. Every 'C' program must contain _____. (K.B+U.B)
(A) Main (B) Link (C) Body of Main () (D) Both B & C

www.ilmkidunya.com

15. 'C' program follows ___ order for execution (K.B)
(A) Sequential (B) Decently (C) Selection (D) None of these

16. 'C' is a ____ language. (K.B+U.B)
(A) Case sensitive (B) Not case sensitive (C) Modular (D) None of these

17. In 'C' language a ____ A (K.B+U.B)
(A) = (B) ≠ (C) sometime (D) None of these

18. Each statement ends with _____. (K.B+U.B)
(A) : (B) ; (C) , (D) #

19. Semicolon(;) is used at the end of ____ statement. (K.B+U.B+A.B)
(A) Every (B) First (C) Last (D) None of these

20. Anything after _____ on the same line is considered a single line comment.(K.B+U.B)
(A) / (B) // (C) * (D) None of these

21. Anything after // on the same line is considered a _____. (K.B+U.B)
(A) Rule (B) Comment (C) Single line comment (D) Multiline comment

22. Multiline comments starts with ____. (K.B)
(A) */ (B) /* (C) // (D) *

23. Anything between /* and */ is considered _____. (K.B)
(A) Very words (B) Comment (C) Reserve words (D) None of these

1.2.3 COMMENTS IN 'C'

LONG QUESTIONS

1. Define Comments in 'C' language program? Explain its types (K.B)

Ans: **PURPOSE AND SYNTAX OF COMMENTS IN 'C' PROGRAMS**

Comments are the statements in a program that are ignored by the compiler and do not get executed. Usually comments are written in natural language e.g. in English language, in order to provide description of our code.

Purpose of writing comments:

Comments can be thought of as documentation of the program. Their purpose is twofold.

1) They facilitate other programmers to understand our code.
2) They help us to understand our own code even after years of writing it.

We do not want these statements to be executed, because it may cause syntax error as the statements are written in natural language.

Syntax of writing comments:

aIn C programming language, there are two types of comments.

1. Single-line Comments
2. Multi-line Comments

Single-line comments start with //. Anything after // on the same line, is considered a comment. For example, // This is a comment.

Multi-line comments start with /* and end at */. Anything between **/*** and ***/** is considered a comment, even on multiple lines. For example,

```
/* this is
a multi-line
comment */
```

Following example code demonstrates the usage of comments:

Example:

```
#include <stdio.h>
/*this program displays "I am a student of class 10" on the output screen*/
void main( )
{ //body of main function starts from here printf ("I am a student of class 10");
} //body of main function ends here.
```


www.ilmkidunya.com

SHORT QUESTIONS

Q.1 Define comments in a 'C' language program. (K.B)

Ans: Comments are the statements in a program that are ignored by the compiler and do not get executed. Usually comments are written in natural language e.g. in English language, in order to provide description of the code. Comments are used to increase the readability of the program.

TYPES OF COMMENT:

- Single-line comments
- Multi-line comments

Q.2 What are single line comments? (K.B)

Ans: Start with //. Anything after // On the same line, is considered to be a comment. For example, //This is a comment.

Example:

// This is a comment.

Q.3 What are multiple comments? (K.B)

Ans: start with /* and end at */. Anything between /* and */ is considered a comment, even on multiple lines. For example,

Example:

/*This is
a multi-line Comment*/

Q.4 How many types of comments can be used in 'C'? (K.B+U.B)

Ans: 1- Single-line comments
2- Multi-line comments

Q.5 Differentiate between single line and multiple comments. (K.B+U.B)

Ans: The differences between single line and multiline comment are as follows:

Single Line Comment	Multiline Comment
• The // is used as single line comment.	• The /*….*/ is used for multiple line comments.
• The syntax of single line comment is: // comment	• The syntax of multiline comment is: /*……..*/
Example: • Comments comprise on one line. • // Hello to the world	**Example:** • Comment can span to multiple lines: • /* Welcome To the C language This is my first program*/

Q.6 Write a program that uses comments. (A.B)

Ans:

EXAMPLE CODE

```
/*This program displays "I am a student of class 10"
on the output screen*/
#include <stdio.h>
void main( )
{ //body of main function starts from here
printf ("I am a student of class 10");
} //body of main function ends here
```


www.ilmkidunya.com

Q.7 **(A.B)**

ACTIVITY 1.5

Tick valid comments among the following:

- *comment goes here*
- **/comment goes here/ → invalid**
- %comment goes here%
- /* comment goes **here*/ → valid**
- /* comment goes here/
- //comment goes here*/

MULTIPLE CHOICE QUESTIONS

1. Which is related to 'C' comments? **(K.B+U.B)**
(A) Single Line (B) Multiple Line (C) Statement Terminator (D) Both 'A' & 'B'

2. It is good programming practice to add in C program: **(K.B)**
(A) Comment/s (B) Question Marks (C) Variable (D) Constant

3. Which one of the following is used to show single line comments: **(K.B+U.B)**
(A) // (B) || (C)\\ (D) !

4. Multi line comment in C language represented as: **(K.B+U.B)**
(A) // (B) */*/ (C) /**/ (D) *//*

5. _____ are statements in a program that are ignored by the compiler. **(K.B)**
(A) Keywords (B) Reserve words (C) Comments (D) None of these

6. Compiler ignores _____. **(K.B)**
(A) Reserve words (B) Comments (C) Keywords (D) None of these

7. Comments are the statement that are _____. **(K.B)**
(A) not executed (B) ignored (C) Both 'A' & 'B' (D) Produced

8. Usually comments are written in _____. **(K.B)**
(A) English Language (B) Natural Language (C) Both 'A' & 'B' (D) None of these

9. Usually _____ are written in natural language. **(K.B)**
(A) Keywords (B) Reserve words (C) Comments (D) None of these

10. Comments provide _____. **(K.B)**
(A) Precaution of code (B) Description of code
(C) Syntax of code (D) All of these

11. There are _____ types of writing comments. **(K.B)**
(A) 2 (B) 3 (C) 4 (D) 5

12. _____ facilitate other programmers to understand own code. **(K.B)**
(A) Keywords (B) Reserve words (C) Comments (D) None of these

13. Comments are not executed as _____ may occur. **(K.B)**
(A) Routine error (B) Syntax error (C) Comments (D) None of these

14. Single-line comments starts with ___. **(K.B+U.B)**
(A) // (B) */ (C) (D) *

15. _____ comments starts with // **(K.B+U.B)**
(A)Multiline comment (B) Single-line (C) Both 'A' & 'B' (D) None of these

www.ilmkidunya.com

1.3 CONSTANTS AND VARIABLES
1.3.1 CONSTANTS
1.3.2 VARIABLES

LONG QUESTIONS

1. Define Constants? Explain its types. (K.B+U.B)

Ans: Constants are the values that cannot be changed by a program e.g. 5, 75.7, 1500 etc. In C language primarily we have three types of constants:

1. **Integer Constants:** These are the values without a decimal point e.g. 7, 1256, 30100, 55555, -54, -2349 etc. They can be positive or negative. If the value is not preceded by a sign, it is considered as positive.
2. **Real Constants:** These are the values including a decimal point e.g. 3.14, 15.3333, 75.0, -1575.76, -7941.2345 etc. They can also be positive or negative.
3. **Character Constants:** Any Single small case letter, upper case letter, digit, punctuation mark, special symbol enclosed within ' ' is considered a character constant e.g. '5', '7', 'a', 'X', '! ', '; ' etc. A digit used as a character constant i.e. 9, is different from a digit used as an integer constant i.e. 9. We can add two integer constants to get the obvious mathematical result e.g. 9 + 8 = 17, but we cannot add a character constant to another character constant to get the obvious mathematical result e.g. '9' + '8' $\neq$ 17.

2. Define variables? Explain its types. (K.B+U.B)

Ans: A variable is a symbolic name that represents a value that can change during execution of a program. A variable has a name, known as variable name and it holds data of other types. A number or any other types of data held in a variable is called its value. It also indicates the types of value variable can represent.

Variables are of two types:

- Numeric Variables
- Character Variables

Numeric Variables:

Numeric variables are used to represent numeric values in computer programs. They represent an integer and floating-point values.

Examples:

- Sum
- Avg
- Length
- Salary
- Marks

Character Variables:

Character variables represent character values in computer programs. It can represent a single character or a string of characters.

Examples:

- Name
- City
- Gender

Explanation:

When a variable is used in a computer program, the computer associates it with a particular memory location. The value of a variable at any time is the value stored in the associated memory location at that time. Variables are used so that the same space in memory can hold different values at different times.

www.ilmkidunya.com

SHORT QUESTIONS

Q.1 Define character set of 'C' language. (K.B)

Ans: Each language has a basic set of alphabets (character set) that are combined in an allowable manner to form words, and then these words can be used to form sentences. Similarly in C programming language we have a character set that includes:

1) Alphabets (A, B, ……, Y, Z), (a, b …. y, z)
2) Digits (0 - 9)
3) Special symbols (~ '!@#% ^ & * ( ) _ - + = | \ { } [ ] : ; " ' < > , . ? /)

Q.2 Define constants of 'C' language. (K.B)

Ans: Constants are the values that cannot be changed by a program e.g. 5, 75.7, 1500 etc. In C language, primarily we have three types of constants:

Types of constant:

- Integer constants
- Real constants
- Character constants

Q.3 Define Integer constant. (K.B)

Ans: These are the values without a decimal point e.g. 7, 1256, 30100, 55555, -54, -2349 etc. They can be positive or negative. If the value is not preceded by a sign, it is considered as positive.

Q.4 Define real constants. (K.B)

Ans: These are the values including a decimal point e.g. 3.14, 15.3333, 75.0, -1575.76, -7941.2345 etc. They can also be positive or negative.

Q.5 Define character constant. (K.B)

Ans: Any single small case letter, upper case letter, digit, punctuation mark, special symbol enclosed within ' ' is considered a character constant e.g. '5', '7', 'a', 'X', '! ', '; ' etc.

Q.6 Differentiate between real constant and integer constant. (K.B + U.B)

Ans:

Real Constant	Integer Constant
• These are the values including a decimal point	• These are the values without a decimal point
• They can also be positive or negative.	• They can be positive or negative. • If the value is not preceded by a sign, it is considered as positive.
Example: • 3.14, 15.3333, 75.0, -1575.76, -7941.2345	**Example:** • 7, 1256, 30100, 55555, -54, -2349

ACTIVITY 1.6 (A.B)

Identify the type of constant for each of the following values.

Solution:

12 → integer constant
1.2 → real constant
'*' → character constant
-21 → integer constant
32.768 → real constant

www.ilmkidunya.com

‘a’ → character constant
-12.3 → real constant
41 → integer constant
40.0 → real constant
‘1’ → character constant

Q.7 Tell a type of constant from the following list. (K.B+U.B+A.B)

12
1.2
‘*’
-21
32.768
‘a’
-12.3
41
40.0
‘1’

Ans

Constant	Type of Constant
12	Integer constant
1.2	Real constant
‘*’	Character constant
-21	Integer constant
32.768	Real constant
‘a’	Character constant
-12.3	Real constant
41	Integer constant
40.0	Real constant
‘1’	Character constant

Q.8 Define variables. (K.B)

Ans: A variable is actually a name given to a memory location, as the data is physically stored inside the computer’s memory. The value of a variable can be changed in a program. It means that, in a program, if a variable contains value 5, then later we can give it another value that replaces the value 5. Each variable has a unique name called identifier and has a data type.

www.ilmkidunya.com

Q.9 Write down any three names of data type. (K.B+U.B+A.B)

Ans:

Type of Data	Matching Data Type in C language	Sample Values
Integer	int	123
Real	float	23.5
Character	char	'a'

Each variable in C language has a data type. The data type not only describes the type of data to be stored inside the variable but also the number of bytes that the compiler needs to reserve for data storage.

MULTIPLE CHOICE QUESIONS

1. Each language has a basic set of alphabets_____. (K.B)
(A) Comments (B) Keyboards (C) Character Set (D) Variable

2. In c programming we have a charter set that includes ____. (K.B+U.B)
(A) Alphabets (B) Digits (C) Special Symbols (D) All of these

3. How many digits in character set are available ranging from _____. (K.B)
(A) 1-9 (B) 0-9 (C) 1-10 (D) 0-10

4. Constant and variable are used in program to write: (K.B)
(A) Expression (B) Equation (C) Statement (D) Semantic

5. The quantities whose value do not change during program execution is called. (K.B)
(A) Variable (B) Constant (C) Reserve word (D) Expression

6. Which one of the following is not a type of constant? (K.B)
(A) Numeric (B) String (C) Reserve Word (D) Both A & B

7. How many types of numeric constant? (K.B)
(A) 2 (B) 4 (C) 3 (D) 5

8. Which one of the following is an example of floating-point numeric constants? (K.B+U.B)
(A) 7145 (B) -234 (C) 166.75 (D) 26

9. String constant are written with in: (K.B)
(A) "" (B) ( ) (C) [ ] (D) { }

10. Which one of the following is an example of character constant is: (K.B+U.B)
(A) 'a' (B) (A) (C) "a" (D) {a}

11. A symbolic name that represents a value that can change during execution of program is called: (K.B+U.B)
(A) Constant (B) Variable (C) String (D) Reserve Word

12. Which one is an example of real constant. (K.B+U.B)
(A) 15.33 (B) -37.8 (C) 37 (D) Both A and B

13. Character constant are always written inside. (K.B)
(A) "" (B) ** (C) , (D) 1,1

14. Sum, avg, length, salary and marks are examples of: (K.B+U.B)
(A) Numeric Constant (B) Data Set
(C) Numeric Variable (D) Character Constant

15. Example of character variable includes: (K.B+U.B)
(A) Name (B) Gender (C) City (D) All of these

www.ilmkidunya.com

16. A variable used in a computer program associates with a: (K.B+U.B+A.B)
(A) Device (B) Constant (C) Memory location (D) Expression

17. In C program, the length of variable name has _____ significant characters. (K.B+U.B)
(A) 21 (B) 31 (C) 30 (D) 32

18. Which one of the following cannot be used as a variable? (K.B+U.B)
(A) Special character (B) Character (C) Digits (D) Variable

19. In C language which of following cannot be used as variable? (K.B+U.B)
(A) auto (B) if (C) int (D) All of these

20. Which one is a correct example of variable name? (K.B+U.B)
(A) Mass (B) Over Time (C) StuID (D) Book ID

21. Each variable has a _____. (K.B+U.B)
(A) Unique Name (B) Identifier (C) Data Type (D) All of these

1.3.3 DATA TYPES OF A VARIABLES

SHORT QUESTIONS

Q.1 Define data types. (K.B)

Ans: Each variable in C language has a data type. The data type not only describes the type of data to be stored inside the variable but also the number of bytes that the compiler needs to reserve for data storage.

- Integer
- Float
- Character

Q.2 Define integer data type. (K.B)

Ans: Integer data type is used to store integer values (whole numbers). Integer takes up 4 bytes of memory. To declare a variable of type integer, we use the keyword int.

Signed int: A signed int can store both positive and negative values ranging from -2, 147, 483, 648 to 2,147, 483, 647. By default. Type int is considered as a signed integer.

Unsigned int: An unsigned int can store only positive values and its value ranges from 0 to +4,292,967,295. Keyword unsigned int is used to declare an unsigned integer.

Q.3 Define signed integer. (K.B)

Ans: A signed int can store both positive and negative values ranging from -2, 147, 483, 648 to 2,147, 483, 647. By default type 'int' is considered as a signed integer.

Q.4 Define unsigned integer. (K.B)

Ans: An unsigned into can store only positive values and its value ranges from 0 to +4,292,967,295. Keyword unsigned int is used to declare an unsigned integer.

Q.5 Define float data type. (K.B)

Ans: Float data type is used to store a real number (number with floating point) up to six digits of precision. To declare a variable of type float, we use the keyword float. A float uses 4 bytes of memory. Its value ranges from 3.4×10^{-38} to 3.4×10^{38}.

Q.6 Define character data type. (K.B)

Ans: To declare character type variables in C, we use the keyword char. It takes up just 1 byte of memory for storage. A variable of type char can store one character only.

www.ilmkidunya.com

Q.7 Differentiate between integer and floats data types. (K.B+U.B)

Ans:

Integer	Float
• Integer data type is used to store integer values (whole numbers).	• Float data type is used to store a real number (number with floating point) up to six digits of precision.
• Integer takes up 4 bytes of memory. • To declare a variable of type integer, we use the keyword **int**.	• A float uses 4 bytes of memory. • To declare a variable of type float, we use the keyword **float**.
Example: • -32,768 to +32,767	**Example:** • 3.4×10^{-38} to 3.4×10^{38}

MULTIPE CHOICE QUESTIONS

1. How many data types provided by 'C' language? (K.B)
(A) 2 (B) 4 (C) 3 (D) 5

2. In 'C' language int and short int data type occupy memory? (K.B)
(A) 1 Byte (B) 2 Bytes (C) 3 Bytes (D) 4 Bytes

3. In 'C' which data types is not used: (K.B)
(A) short int (B) float (C) unsigned int (D) float int

4. Float data type in 'C' occupy memory space: (K.B)
(A) 2 Bytes (B) 4 Bytes (C) 8 Bytes (D) 70 Bytes

5. In 'C' language a character variable can only store one. (K.B)
(A) Digit (B) Character (C) Byte (D) Word

6. In 'C' language unsigned integer data type have range. (K.B)
(A) 0-65535 (B) $10.^{0888}$-10^{308} (C) 0-4,294,697,295 (D) 0-32768

7. In 'C' language single integer data type have range. (K.B)
(A) -22,147,483,648 102,147,483,647 (B) -327651032765
(C) 0.56655 (D) -2,147,483,648 – 2,147,483647

8. Character data type occupy memory space. (K.B)
(A) 3 bytes (B) 1 byte (C) 4 bytes (D) 0 byte

1.3.4 NAME OF A VARIABLE

LONG QUESTIONS

1. Explain rules for naming variables in 'C' language. (K.B+U.B)

Ans: Rules For Naming Variables in 'C' Language

The following are the rules for specifying variable names in C language.

- A variable begins with a letter or underscore and may consist of **letters, underscores "_" and/or digits.**
- The underscore may be used to improve readability of the variable name. For example, over_time.
- There is no restriction on the length of a variable name. However, only the first 31 characters of a variable are significant. This means that if two variables have the same first 31 characters, they are considered to be the same variables.
- Both upper and lower-case letters are allowed in naming variables. An upper-case letter from Avg or avg.
- Special characters cannot be used as variable name e.g, #, ?, @, etc.
- Reserved words of C language such as int, case, if, etc, cannot be used as variable name.
- There must be no embedded blank in the name of variable. E.g father's name.

www.ilmkidunya.com

SHORT QUESTIONS

Q.1 Write down any three rules for naming a variable in 'C' language (K.B)

Ans: Each variable must have a unique name or identifier. Following rules are used to name a variable.

1. A variable name can only contain alphabets (uppercase or lowercase), digits and underscore (_)sign
2. Variable name must begin with a letter or an underscore, it cannot begin with a digit.
3. A reserved word cannot be used as a variable name.

Q.2 ACTIVITY 1.7 (A.B)

ACTIVITY 1.7				
Encircle the valid variable names among the following:				
Hello,	1var	roll_num → valid	Air23Blue→valid	float
Case	$car	name → valid	=color	Float

MULTIPLE CHOICE QUESTION

1. A variable name can only contain. (K.B)
(A) Digits (B) Alphabets (C) Underscore (D) All of these

2. We should give appropriate name to a variable that desirable its ____. (K.B)
(A) Evolution (B) Purpose (C) Order (D) Type

1.3.5 VARIABLE DECLARATION

SHORT QUESTIONS

Q.1 Define variable declaration. (K.B)

OR

What is variable declaration?

Ans: We need to declare a variable before we can use it in the program. Declaring a variable includes specifying its data type and giving it a valid name. Following syntax can be followed to declare a variable.

Some examples of valid variable declarations are as follows:

unsigned int age;
float height;
int salary;
char marital_status;

MULTIPLE CHOICE QUESITONS

1. We need to declare a variable _____ it use. (K.B+U.B)
(A) After (B) Before (C) Compile (D) None of these

2. Declaring a variable includes specifying its_____. (K.B)
(A) Data Type (B) Valid Name (C) Both A & B (D) None of these

3. Multiple variable of _____ data type many also be declared in a single standard. (K.B)
(A) Different (B) Same (C) Both A & B (D) None of These

4. A variable cannot be declared unless we mention its _____. (K.B)
(A) Format (B) Data Type (C) Range (D) None of these

5. After declaring a variable its data type _____ be changed. (K.B+U.B)
(A) Can (B) Cannot (C) Both A & B (D) None of these

www.ilmkidunya.com

1.3.6 VARIABLE INITIALIZATION

SHORT QUESTIONS

Q.1 Define variable initialization. (K.B)

OR

What is variable initialization?

Ans: Assigning value to a variable for the first time is called variable initialization. **C** language allows us to initialize a variable both at the time of declaration, and after declaring it. For initializing a variable at the time of declaration, we use the following general structure.

data_type variable_name = value;

Q.2 What is the difference between variable declaration and variable initialization?(K.B + U.B)

Ans:

Variable Declaration:	Variable Initialization

We need to declare a variable before we can use it in the program. Declaring a variable includes specifying its data type and giving it a valid name.
following syntax can be followed to declare a variable.
data_type variable_name;
Example:
Some examples of valid variable declarations are as follows:
unsigned int age;
float height;
int salary;
char marital_status;
Multiple variables of same data type may also be declared in a single statement, as shown in the following examples:
unsigned int age, basic_salary,gross_salary;
int points_scored, steps;
	Assigning value to a variable for the first time is called variable initialization. C language allows us to initialize a variable both at the time of declaration, and after declaring it.
	For initializing a variable at the time of declaration, we use the following general structure.
	data_type variable_name = value;
	Example:
	Following example shows a program that demonstrates the declaration and initialization of two variables.
	#include<stdio.h>
	void main ()
	{
	char grade; //Variable grade is declared
	int value = 25; /*Variable value is declared and initialized.*/
	grade = 'A'; //Variable grade is initialized
	}

www.ilmkidunya.com

Q.3 Write a program that declares variables of appropriate data types to store your personal data. Initialize these variables with the following data: (A.B)

- **Initial letter of your name**
- **Initial letter of your gender**
- **Your age**
- **Your marks in 8th class**
- **Your height**

Ans:

ACTIVITY 1.8

Solution

```
#include < stdio.h>
#include < conio.h>
void main ()
{
char name = 'A';       // initial letter of your name
char gender = 'M';     // initial letter of your gender
int age = 26;          // your age
int marks = 500;       // marks in 8th class
float height =5.8;     // your height
}
```

MULTIPLE CHOICE QUESTIONS

1. Assigning value to a variable for the first time is called variable. (K.B)

(A) Declaration (B) Initialization (C) Compilation (D) None of these

2. For initializing a variable at the time of declaration, we use: (K.B)

(A) Data Type (B) Variable Name (C) Both A & B (D) None of these

www.ilmkidunya.com

EXERCISE

Q.1 Multiple Choice Questions

1) A software that facilitates programmers in writing computer program known as_____.(K.B)
(A) a complier (B) an editor (C) an IDE (D) a debugger

2) _____is a software that is responsible for the conversion of program files to machine understandable and executable code. (K.B+U.B)
(A) Compiler (B) Editor (C) IDE (D) Debugger

3) Every programming language has some primitive building blocks and follows some grammar rules known as its _____. (K.B+U.B)
(A) Programming Rules (B) Syntax (C) Building Blocks (D) Semantic Rules

4) A list of words that are predefine and must not be used by the programmer to name his own variables are known as _____. (K.B+U.B)
(A) Auto Words (B) Reserved Words (C) Restricted Words (D) Predefined Words

5) Include statements are written in _____ section. (K.B)
(A) Header (B) Main (C) Comments (D) Print

6) _____ are added in the source code to further explain the techniques and algorithms used by the programmer. (K.B)
(A) Messages (B) Hints (C) Comments (D) Explanations

7) _____ are the values that do not change during the whole execution of program. (K.B)
(A) Variables (B) Constants (C) Strings (D) Comments

8) A float uses___ bytes of memory. (K.B)
(A) 3 (B) 4 (C) 5 (D) 6

9) For initializing variable, we use ____ operator. (K.B)
(A) → (B) = (C) @ (D)?

10) _____ can be thought of as a container to store constants. (K.B)
(A) Box (B) Jar (C) Variable (D) Collection

ANSWER KEY

1	C	6	C
2	A	7	B
3	B	8	B
4	B	9	B
5	A	10	C

Q.2 True of False

1) An IDE combines text editors, libraries, compilers and debuggers in a single interface. **(K.B) T/F**
2) Computers require the conversion of the code written in program file to machine language in order to execute it. **(K.B+U.B) T/F**
3) Column in reserved word in C programming language. **(K.B) T/F**
4) *comment goes here* is a valid comment. **(K.B) T/F**
5) Float can store a real number upto six digits or precision. **(K.B) T/F**

Q.3 Define the following. (K.B)

1) IDE
Ans: A software that provides a programming environment, which facilitates the programmer in writing and executing computer programs, is known as an Integrated Development Environment (IDE).

2) Compiler
Ans: A compiler is a software that is responsible for conversion of a computer program written in some programming language to machine language code and vice versa.

www.ilmkidunya.com

3) Reserved Words

Ans: Every programming language has a list of words that are predefined. Each word has its specific meaning already known to the compiler. These words are known as reserved words or keywords. If a programmer gives them a definition of his own, it causes a syntax error. Table shows the list of reserved words in C programming language. There are 32 reserved words in C.

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

4) Main section of a program

Ans: It consists of a main () function. Every C program must contain a main () function and it is the starting point of execution.

5) char data type

Ans: To declare character type variables in C, we use the keyword char. It takes up just 1 byte of memory for storage. A variable of type char can store one character only.

Q.4 Briefly answer the following questions. (K.B+U.B)

1) Why do we need a programming environment?

Ans: In order to correctly perform any task, we need to have proper tools. For example, for gardening we need gardening tools and for painting we need a collection of paints, brushes and canvas. Similarly, we need proper tools for programming. A collection of all the necessary tools for programming makes up a programming environment. It is essential to setup a programming environment before we start writing programs. It works as a basic platform for us to write and execute programs.

2) Write the steps to create a C program file in the IDE of your lab computer.

Ans: A software that provides a programming environment to facilitate programmers in writing and executing computer programs is known as an Integrated Development Environment (IDE). An IDE has a graphical user interface (GUI), meaning that a user can interact with it using windows and buttons to provide input and get output. An IDE consists of tools that help a programmer throughout the phases of writing, executing and testing a computer program. This is achieved by combining text editors, compilers and debuggers in a single interface. Some of the many available IDEs for C programming language are:

1. Visual Studio
2. Xcode
3. Code::Bocks

A text editor is a software that allows programmers to write and edit computer programs. All IDEs have their own specific text editors. It is the main screen of an IDE where we can write our programs. Computers only understand and work in machine language consisting of 0s and 1s. They require the conversion of a program written in programming language to machine language in order to execute it. This is achieved using a compiler. A compiler is a software that is responsible for conversion of a computer program written in some high level programming language to machine language code.

www.ilmkidunya.com

3) Describe the purpose of a compiler.

Ans: Computers only understand and work in machine language consisting of 0s and 1s. They require the conversion of a program written in programming language to machine language, in order to execute it. This is achieved using a compiler. A compiler is a software that is responsible for conversion of a computer program written in some high level programming language to machine language code.

4) List down five reserve words in C programming language.

Ans: The five reserved words are as follows

(i) auto (ii) double (iii) int (v) if

C language has total 32 reserved words

Reserved Words in 'C' language

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

5) Discuss the main parts of the structure of a 'C' program.

Ans: **Structure of a 'C' Program**

A 'C' program can be divided into three main parts:

1. **Link section or header section:** While writing programs in 'C' language, we make extensive use of functions that are already defined in the language. But before using the existing functions, we need to include the files where these functions have been defined. These files are called header files. We include these header files in our program by writing the include statements at the top of program.
2. **Main section of a program:** It consists of main ( ) function. Every 'C' program has a main function and it is the starting point of execution.
3. **Body of main () function:** The body of main () is enclosed in the curly braces {}. All the statements inside the curly braces make the body of main function. The statement printf ("Hello World!"); uses a predefined function *printf* to display the statement Hello World! on computer screen.

6) Why do we use comments in programming?

Ans: **Purpose of Comments in C Programs**

Comments are the statements in a program that are ignored by the compiler and do not get executed. Usually comments are written in natural language e.g. in English language in order to provide description of our code.

Types:

1. Single- line comment //
2. Multi- line comment /* */

www.ilmkidunya.com

7) Differentiate between constants and variables.

Ans:

CONSTANTS	VARIABLES
Each language has a basic set of alphabets (character set) that are combined in an allowable manner to form words, and then these words can be used to form sentences. Similarly, in C programming language we have a character set that includes: Alphabets (A, B, ……, Y, Z), (a, b …. y, z) Digits (0 – 9) Special symbols (~ '!@#% ^ & * ( ) _ - + = | \ { } [ ] : ; " ' < > , . ? /) These alphabets, digits and special symbols when combined in an allowable manner, form constants, variables and keywords (also known as reserved words). We have already discussed the concept of reserved words. In the following, we discuss the concept of constants and variables. **Constants:** Constants are the values that cannot be changed by a program e.g. 5, 75.7, 1500 etc. In C language, primarily we have three types of constants: **Integer Constants:** These are the values without a decimal point e.g. 7, 1256, 30100, 55555, -54, -2349 etc. They can be positive or negative. If the value is not preceded by any sign, it is considered as positive. **Real Constants:** These are the values including a decimal point e.g. 3.14, 15.3333, 75.0, -1575.76, -7941.2345 etc. They can also be positive or negative. **Character Constants:** Any Single small case letter, upper case letter, digit, punctuation mark, special symbol enclosed within ' ' is considered a character constant e.g. '5', '7', 'a', 'X', '! ', '; ' etc.	A variable is actually a name given to a memory location, as the data is physically stored inside the computer's memory. The value of a variable can be changed in a program. It means that, in a program, if a variable contains value 5, then later we can give it another value that replaces the value 5. Each variable has a unique name called identifier and has a data type. Data type describes the type of data that can be stored in the variable. C language has different data types such as int, float, and char. The types int, float and char are used to store integer, real and character data respectively. Table 1.2 shows the matching data types in C language, against different types of data. **Type of Data** | **Matching Data Type in C language** | **Sample Values** Integer | int | 123 Real | float | 23.5 Character | char | 'a'

8) Write down the rules for naming variables.

Ans:

1. A variable name can only contain alphabets (upper case or lowercase), digits and underscore sign.
2. Variable name must begin with a letter or an underscore, it cannot begin with a digit.
3. A reserved word cannot be used as a variable name.
4. There is no strict rule on how long a variable name should be, but we should choose a concise length for variable name to follow good design practice.
   Some examples of valid variable names are height, Average Weight, _var1.

www.ilmkidunya.com

9) Differentiate between char and int.

Ans:

Integer – int (signed/unsigned)	Character-char
Integer data type is used to store integer values (whole numbers). Integer takes up 4 bytes of memory. To declare a variable of type integer, we use the keyword int. **Signed int:** A signed int can store both positive and negative values ranging from -2, 147, 483, 648 to 2,147, 483, 647 By default. Type int is considered as a signed integer. **Unsigned int:** An unsigned into can store only positive values and its value ranges from 0 to +4,292,967,295. Keyword unsigned int is used to declare an unsigned integer.	To declare character type variables in C, we use the keyword char. It takes up just 1 byte of memory for storage. A variable of type char can store one character only.

10) How can we declare and initialize a variable?

Ans:

Variable Declaration	Variable Initialization

We need to declare a variable before we can use it in the program. Declaring a variable includes specifying its data type and giving it a valid name. Following syntax can be followed to declare a variable.
data_type variable name;
Some examples of valid variable declarations are as follows:
unsigned int age;
float height;
int salary;
char marital_status;
Multiple variables of same data type may also be declared in a single statement, as shown in the following examples: unsigned int age, basic_ salary, gross_ salary;
int points_ scored, steps;
float height, marks;
char marital_status, gender;
A variable cannot be declared unless we mention its data type. After declaring a variable, its data type cannot be changed. Declaring a variable specifies the type of variable, the range of values allowed by that variable, and the kind of operations that can be performed on it. Following example shows a program declaring two variables:
EXAMPLE CODE
void main ()
{
char grade;
int value;
}
	Assigning value to a variable for the first time is called variable initialization. C language allows us to initialize a variable both at the time of declaration, and after declaring it. For initializing a variable at the time of declaration, we use the following general structure.
	data_type variable name = value;
	Following example shows a program that demonstrates the declaration and initialization of two variables.
	example code
	#include<stdio.h>
	void main ()
	{
	char grade; //Variable grade is declared
	int value = 25; /*Variable value id declared and initialized. */
	grade = 'A'; //Variable grade is initialized
	}

www.ilmkidunya.com

PROGRAMMING EXERCISE

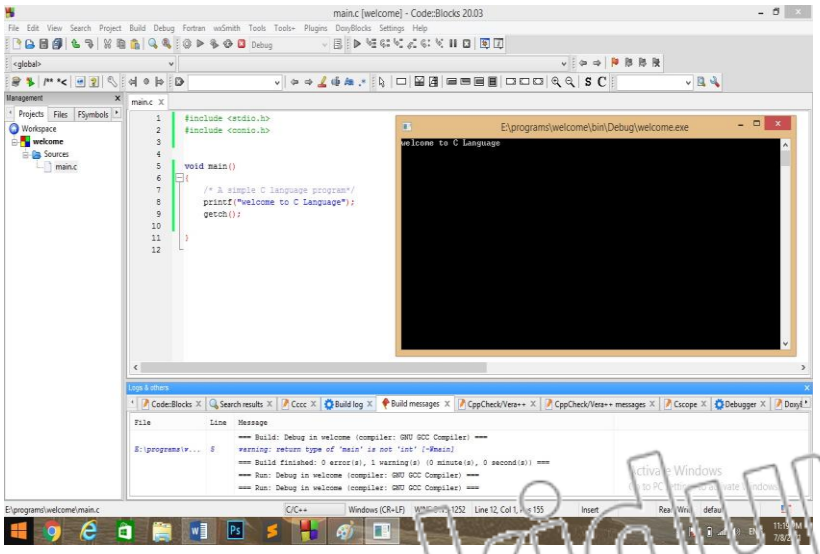
Exercise 1

- **With the help of your teacher open the IDE installed on your lab computer for writing C programs.**
- **Write the following program in the editor and save it as "welcome.c".**

```
#include <stdio.h>
#include <conio.h>
void main ()
{
//A simple C language program
printf ("Welcome to C language");
getch ( );
}
```

Run the program to see Welcome to C language printed on the screen as output.Solution:

```
#include <stdio.h>
#include <conio.h>
void main()
{
  /* A simple C language program*/
   printf("welcome to C Language");
   getch();
}
```



www.ilmkidunya.com

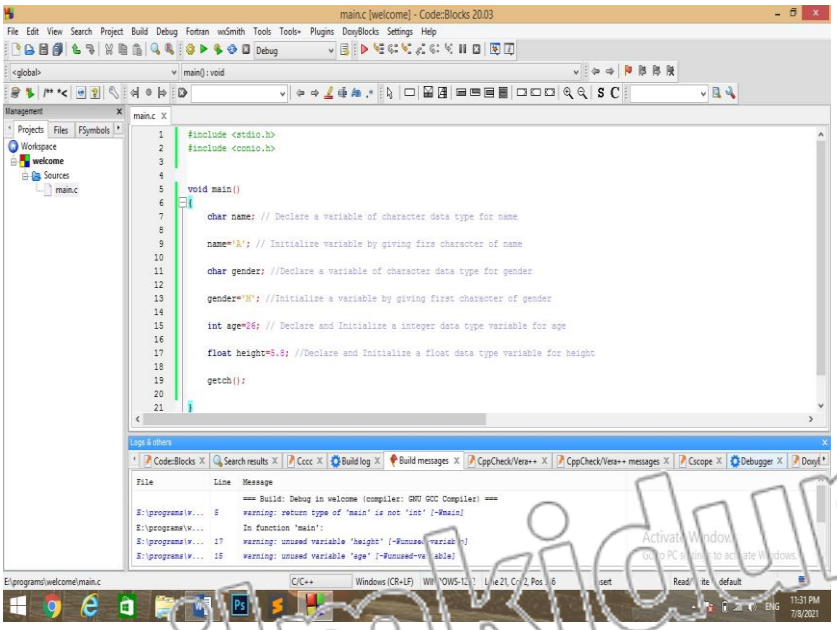
Exercise 2

Write a program that declares variables of appropriate data types to store the personal data about your best friend. Initialize these variables with the following data:

- **Initial letter of his name**
- **Initial letter of his gender**
- **His age**
- **His height**

Solution:

```
#include <stdio.h>
#include <conio.h>
void main()
{
char name; // Declare a variable of character data type for name
name='A'; // Initialize variable by giving firs character of name
char gender; //Declare a variable of character data type for gender
gender='M'; //Initialize a variable by giving first character of gender
int age=26; // Declare and Initialize a integer data type variable for age
float height=5.8; //Declare and Initialize a float data type variable for height
getch();
}
```



www.ilmkidunya.com

ANSWER KEY

1.1 PROGRAMMING EVIRONMENT

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A	D	B	A	D	A	A	D	C	D	A	D	A	A	C	A
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
A	D	C	A	A	D	D	D	A	B	A	A	A	A	A	B

1.2 PROGRAMMING BASIC

1	2	3	4
A	A	A	A

1.2.1 RESERVED WORDS

1	2	3	4
D	B	D	C

1.2.2 STRUCTURE OF C PROGRAM

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
D	D	B	D	A	B	D	B	C	C	A	A	A	D	A	A
17	18	19	20	21	22	23									
B	B	A	B	B	B	B									

1.2.3 COMMENTS IN C

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
D	A	C	C	C	B	C	B	C	B	A	C	B	D	A	B

1.3 CONTANTS AND VARIABLE

1.3.1 CONTANTS

1.3.2 VARIABLES

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
C	D	B	A	B	C	C	C	A	A	B	D	A	C	D	C
17	18	19	20	21											
B	A	D	A	D											

1.3.3 DATA TYPES OF A VARIABLE

1	2	3	4	5	6	7	8
C	D	D	B	C	C	D	B

1.3.4 NAME OF A VARIABLE

1	2
D	B

1.3.5 VARIALE DECLARATION

1	2	3	4	5
B	C	B	B	B

1.3.6 VARIABLE INTILIZATION

1	2
B	C

www.ilmkidunya.com