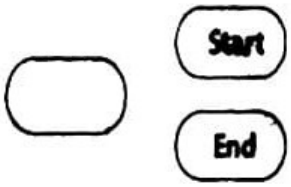
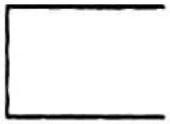


Q.19:- What is the use of Rectangle symbol?

Ans.:- Use of Rectangle Symbol

Rectangle symbol is used to represent the process or action taken in a flowchart.



X = 15

Unit-5

Computer Programming

EXERCISE

Q1. Tick the correct choice for the following questions.

i. **A sequence of instructions given to the computer to perform a specific task is called?**
a. Data b. Program
c. Programming d. Information

ii. **What is a named space in the computer's memory whose value can be changed during the execution of a program?**
a. Variable b. Constant
c. Program d. Data

iii. **In BASIC programming language which character is used as a last character with string variable?**
a. ? b. & c. % d. $

iv. **Which shortcut key is used to apply RUN command in GWBASIC?**
a. F1 b. F2 c. F3 d. F4

v. **Which of the following error cannot be detected by a computer?**
a. Syntax error b. Run time error
c. Logical error d. Execution error

vi. **Which of the following is a logical operator?**
a. + b. - c. > d. /

vii. **Which of the following operator has the highest priority?**
a. + b. ^ c. - d. *

viii. **If A=5, B=3 and C=2, what will be the answer of the following expression? Exp = A + B * C**
a. 10 b. 16 c. 11 d. 13

ix. **Which of the following statement is used to accept data from the user during program execution?**
a. PRINT b. LOAD
c. READ-DATA d. INPUT

x. **The set of rules that define the combination of symbols used by programming languages iscalled_____ .**
a. Syntax b. Data
c. Program d. Logic

Answers:

(i) Program **(ii)** Variable **(iii)** $
(iv) F2 **(v)** Logical error
(vi) > **(vii)** ^ **(viii)** 11
(ix) INPUT **(x)** Syntax

Q2. Fill in the blanks.

i. _____ languages are the means of communication between users and the computer.

ii. _____ are the quantities whose values cannot be changed during program execution or running.

iii. In a program the occurrence of incorrect or unexpected result is called _____.

iv. A _____ error is an error resulting in wrong answer due to programmers own mistake.

v. _____ are symbols that represent particular actions in an expression.

vi. An operator which is used to assign a

value to a variable is called _____.

vii. BASIC stands for _____.

viii. In BASIC language program line number is a positive _____.

ix. _____ are key words which are used to issue instructions to the computer to perform specific tasks.

x. In GWBASIC _____ command is used to clear the screen.

Answers:

(i) Programming **(ii)** Constants
(iii) Error **(iv)** Logical
(v) Operators
(vi) Assignment operator
(vii) Beginner's All purpose Symbolic Instruction Code **(viii)** Integer
(ix) Commands **(x)** CLS

Q3.Match Column A with Column B.

Column A	Column B	Answers
Constants	MOD	Not changeable Quantities
Variables	READ-DATA	Changeable Quantities
Arithmetic Operator	SAVE	MOD
Relational Operator	Changeable Quantities	< >
Command	< >	SAVE
Statement	Not changeable Quantities	READ-DATA

Q4.Give short answers to the following questions:

i. What is a program? Give few examples of programs.

Ans.:- Program:

A computer program is a sequence of instructions given to the computer to perform a specific task. Programs are written in languages called computer languages or programming languages. Computer can only do exactly what the program tells the computer to do. Computer follows the instructions written in the program to perform a particular task. A computer program is stored as a file on the computer's storage medium, like hard disk.

Example: Few examples of programs are:

- A program to find the average marks of a student in the final exam.
- A program to find area of a circle for given radius.
- A program to solve a quadratic equation.
- A program to control financial matters of a company.
- A program to manage items in a grocery shop.

ii. What is the purpose of a programming language?

Ans.:- Programming Language:

Natural languages, like Urdu, English, Arabic and French, etc. are the means of communication among people to share their ideas and views. In the same way programming languages are the means of communication between users and the computer. Users as well as computers understand these languages.

A programming language is a language designed to develop programs or instructions to communicate with the computer to solve various problems. There are different programming languages having their own set of rules and regulations that the computer programmers follow to write computer programs. Most of these languages use English words as statements and commands to create programs. Some examples of programming languages are FORTRAN, COBOL, C, C++, C#, Java and BASIC.

iii. Differentiate between a constant and a variable.

Ans.:-Constants and Variables:

Constants and variables are the two important terminologies used in programming.

a) Constants:

Constants are the quantities whose values cannot be changed during program execution or running. These are the named spaces in the computer's memory. Constants are classified as string constants and numeric constants.

b) Variable:

It is a named space in the computer's memory whose value can be changed during the execution of a program. Variables give meaningful names to constants. They are also used to store data and results during program execution. There are two types of variables. Such as numaric and string variable.

iv. Differentiate between Syntax error and logical error.

Ans.:-

Syntax Errors	Logical Errors
The syntax of a programming language is the set of rules that define the combination of symbols used by that language. A syntax error occurs when the instructions written in a program do not follow these rules of the programming language. Syntax errors are easy to find and correct because the computer finds them for the user. Examples: PRNT instead of PRINT 5 = X instead of X = 5	A logical error is an error resulting in wrong answer due to programmers own logical mistake. These errors occur due to wrong use of formulae or providing wrong value to a variable. If a programmer writes a statement that is logically incorrect, the computer will understand and execute it but the result will be wrong. For example, if a programmer accidentally multiplies two variables when he or she meant to divide them, the program will give an incorrect result, but no error message. Such errors cannot be detected by a computer therefore they are hard to find and correct. Examples: SUM = X - Y instead of SUM = X + Y AREA = Pie * R^3 instead of AREA = Pie * R^2

v. What are the rules for defining/declaring variables in GWBASIC?

Ans.:- Rules for defining/declaring variables in GWBASIC:

In programming languages, like BASIC, certain rules are followed to declare/define a variable name. These are:

- Alphabet and numbers can be used for variables.
- The first character of the variable should be an alphabet.
- No special symbol is allowed except Underscore (_).
- Underscore (_) cannot be used as a first or last character.

vi. Name different types of constants with examples.

Ans.:- Different types of constants

Constants are the quantities whose values cannot be changed during program execution or running. Constants are classified as string constants and numeric constants.

- **String Constants:** These are sequence of alphabetic or alphanumeric characters enclosed in double quotation marks. For example "Male", "Married", "Haris", "Pakistan", "H.No. 107", etc.
- **Numeric Constants:** These are

numbers. For example 117, 20.50, -50 etc.

vii. Give the precedence of arithmetic operators?

Ans.:- Precedence of arithmetic operators:

Precedence is the priority that is followed while applying these operators. This becomes important when more than one kind of arithmetic operator appears within one expression. The precedence of arithmetic operators is given in the following table:

Priority	Operator	Symbol
First	Exponentiation operator	^
Second	Multiplication or division operator	* OR /
Third	Mod operator	MOD
Fourth	Addition or Subtraction operator	+ OR –

Same priority operators are resolved from left to right.

In an expression the operations within parenthesis ( ) are resolved first. The usual order inside the parenthesis is maintained as given in the above table.

viii. What is purpose of modulus operator (MOD)?

Ans.:- Purpose of Modulus Operator (MOD):

Modulus (MOD) operator gives the remainder after division.

Example:

35 MOD 6

Output

5

ix. What is the purpose of the following GWBASIC commands?

a) LIST b) RUN

c) LOAD d) SAVE

Ans.;- a) LIST Command:

List command shows the list of all or part of a loaded program on the screen. F1 key is used as a shortcut key for LIST command.

Syntax: LIST [line number][-line number] OR Press F1key

b) RUN:

RUN command is used to execute (get result) of the program. F2 key is used as a shortcut key for RUN command.

Syntax:RUN OR Press F2 key

c) LOAD:

Load command is used to load (open) a file from any storage medium like Hard Disk into the main memory of the computer. F3 key is used as a shortcut key for LOAD command.

Syntax:

LOAD 'filename' OR Press F3 key

d) SAVE:

SAVE command is used to save the program in the Hard Disk or any other storage media. F4 key is used as a shortcut key for SAVE command.

Syntax:

SAVE filename OR Press F4 key

Q5. Give detailed answers to the following questions.

i. Explain different types of errors in programming with examples.

Ans.:- Different Errors in programming:

In a program the occurrence of incorrect or unexpected result is called an error. Errors occur due to some mistake in performing operations in a program. There are two common types of errors in programming.

- Syntax Errors
- Logical Errors

a) Syntax Errors:

A syntax error occurs when the instructions written in a program do not follow these rules of the programming language. Syntax errors are easy to find and correct because the computer finds them for the user.

Examples:

PRNT instead of PRINT

5 = X instead of X = 5

b) Logical Error:

A logical error is an error resulting in wrong answer due to programmers own logical mistake. These errors occur due to wrong use of formulae or providing wrong value to a variable. If a programmer writes a statement that is logically incorrect, the computer will understand and execute it but the result will be wrong. For example, if a programmer accidentally multiplies two variables when he or she meant to divide them, the program will give an incorrect result, but no error message. Such errors cannot be detected by a computer therefore they are hard to find and correct.

Examples:

SUM = X - Y instead of SUM = X + Y

AREA = Pie * R^3 instead of AREA= Pie * R^2

ii. What is arithmetic expression? Explain different types of operators with examples.

Ans.:- Arithmetic Expression:

An expression is a combination of symbols and operators that represents a value. Every expression consists of at least one operand and can have one or more operators. Operands are values, whereas operators are symbols that represent particular actions. For example, in the expression X + 3 X and 3 are operands, and + is an operator.

An expression which represents a numeric value is called an Arithmetic Expression. An Arithmetic Expression is evaluated by performing a sequence of arithmetic operations to obtain a numeric value.

Some examples of Arithmetic Expressions are as follows.

Exp 1 = (A+3)*(C+2)

Exp 2 = 2*X + 3*Y

Exp 3 = 2 * 3.14*R

Algebraic expression cannot be used directly. in programming. It must be converted into computer understandable expression. Few examples are given in the following table.

Algebraic Expression	BASIC Expression
A = L' B	A = L * B
P = 2 (L + B)	P = 2* (L+B)
I = (P×T×R) /100	I = (P* T* R)/100
V = 4/3 Pie R^3	V = 4/3* Pie * R^3

Operators:

Operators are symbols that indicate the type of operation to be performed on the data. There are three common types of operators.

- Arithmetic operators (+, -, *, /, ^, MOD)
- Assignment operator (=)
- Relational operators (<, >, < >, <=, >=)

a) Arithmetic Operators:

Arithmetic Operators are used to perform mathematical calculations like addition, subtraction, division, multiplication and exponentiation. The following table gives the description of arithmetic operators with examples:

Operation	Description	Examples
+	Addition	Gives the sum of values	A + B + C, X + 50, 67 + 45
-	Subtraction	Gives the difference of values	X - 56, A - C, 80 - 56
*	Multiplication	Gives the product of values	A * B, Y * 50, 45 * 60
/	Division	Gives the quotient	X / Y, B / 40, 125 / 25
^	Exponentiation	Raise the value to the power of an exponent	A ^ 10, 5 ^ 2, A ^ B
MOD	Modulus	Gives the remainder after division	35 MOD 6, X MOD Y

Precedence of Arithmetic Operators:

Precedence is the priority that is followed while applying these operators. This becomes important when more than one kind of arithmetic operator appears within one expression. The precedence of arithmetic operators is given in the

following table.

Priority	Operator	Symbol
First	Exponentiation operator	^
Second	Multiplication or division operator	* OR/
Third	Mod operator	MOD
Fourth	Addition or Subtraction operator	+ OR –

Same priority operators are resolved from left to right.

In an expression the operations within parenthesis ( ) are resolved first. The usual order inside the parenthesis is maintained as given in the above table. The following are few examples to show the precedence of arithmetic operators.

Example1: Solve the expression, X + Y / 10, where X = 30 and Y = 50

= x+y/10
= 30 + 50/10
= 30 + 5
= 35

Example 2: Solve the expression, (A+B)* B / 10, where A = 20 and B = 30

= (A+B) * B/10
= (20 + 30) * 30/10 (The operation in the parenthesis will be resolved first)
= 50 * 30/10 (Multiplication will be done next)
= 1500/10 (Division will be resolved in the end)
= 150

b) Assignment operator:

An operator which is used to assign a value to a variable is called Assignment Operator. In programming languages "Equal sign" (=) is used as an assignment operator.

Example 1: A = 5

In the above example 5 is assigned to the variable A.

Example 2: X = A + 2.54

This expression instructs the computer to calculate A+2.54 and assign the result to X.

c) Relational operators:

Relational Operators are used to perform comparisons on two values. The result of comparison is either true (non zero) or false (zero). The following table gives the description of relational operators with examples.

Operator Symbol	Operation	Description	Example
=	Equal to	Returns true if the two values are equal, and false if not.	A = B
<>	Not Equal to	Returns true if the two values are not equal, and false if they are equal.	5 <> 7
>	Greater than	Returns true if the first number is greater than the second, and false if not.	15 > 11
<	Less than	Returns true if the first number is less than the second, and false if not.	7 < 9
>=	Greater than or Equal to	Returns true if the first number is greater than or equal to the second, and false if not.	X+1 >= 7
<=	Less than or Equal to	Returns true if the first number is less than or equal to the second, and false if not.	Y <= 5

iii. Convert the following algebraic expressions into computer expressions.

a. (AB) + (BC)
b. 5X - 7Y
c. A ÷ B x C3
d. B2 - 4AC
e. XY ÷ (X - Y)

Ans.:- a) (AB) + (BC)
=(A*B)+(B*C)

b) 5X - 7Y
=5*X-7*Y

c) A ÷ B x C3
=A/B*C*3

d) B2 - 4AC
=B*2-4*A*C

e) XY ÷ (X - Y)
=X*Y/(X-Y)

iv. Find solution for the following expressions, if A=3, B=4 and C=5

a. C - A * B - 5

b. (A + B) - C^2
c. (B + C) / A * 10
d. A^2 + B^2 - 2*A*B
e. C MOD A * B + C

Ans.:- a) C - A * B - 5
=5-3*4-5
(Multiplication will be solved first)
=5-12-5
(Subtraction will be solved next)
=-7-5
(Addition will be solved in the end)
=-12

b) (A + B) - C^2
=(3+4)-5^2
(Exponentiation will be solved first)
=(3+4)-25
(The operation in the parenthesis will be solved next)
=7-25
(Subtraction will be solved in the end)
=-18

c) (B + C) / A * 10
=(4+5)/3*10
(The operation in the parenthesis will be solved first)
=9/3*10
(Division will be solved next)
=3*10
(Multiplication will be solved in the end)
=30

d) A^2 + B^2 - 2*A*B
=3^2+4^2-2*3*4
(Exponentiation will be solved first)
=9+4^2-2*3*4
(Exponentiation will be solved next)
=9+16-2*3*4
(Multiplication will be solved next)
=9+16-24
(Addition will be solved next)
=25-24
(Subtraction will be solved in the end)
=1
if A=3, B=4 and C=5

e) C MOD A * B + C
=5 MOD 3*4+5
(Multiplication will be solved first)
=5 MOD 12+5
(Mod operator will be solved next)
=

v. Explain the purpose of the following BASIC statements with their syntax and examples.

a. PRINT b. INPUT c. READ and DATA d. IF - THEN -ELSE

Ans.:- a) PRINT Statement:
The PRINT statement is used to display message or output to the screen. As a shortcut "?" can be used for PRINT command.
shortcut "?" can be used for PRINT command.

Syntaxes of PRINT statement.

Syntax 1: Line Number PRINT "Sequence of characters/String"

Example: 10 PRINT "I Love Pakistan"
The above statement will output a string message "I Love Pakistan" on the screen.

Syntax 2: Line Number PRINT "Variables/Constants"

Example: 10 PRINT X,Y,Z
The above statement will output values of variable X, Y and Z on the screen.
20 PRINT 205, -101
The above statement will output constant values 205 and -101 on the screen.

Syntax 3: Line Number PRINT "String",Variables/Constants

Example: 10 PRINT "Sum of numbers =", SUM
The above statement will output a string "Sum of numbers =" with the value of variable SUM on the screen.

Use of Semicolon (;) with PRINT statement:
In PRINT statement, semicolon (;) between strings/variables/constants causes them to be printed right next to each other

without any space between them.

Use of Comma (,) with PRINT statement:

In PRINT statement, comma (,) between strings/variables/constants causes them to be printed with equal spaces between them. Maximum of five values can be printed on single line using commas.

The following program shows the difference between semicolon and comma.

Programme:	Output
10 PRINT "BOOK" ; "Urdu"; "Grade 8"	Book Urdu Grade 8
20 PRINT 123;456;789	123456789
30 PRINT "Book", "Urdu", "Grade 8"	Book Urdu Grade 8
40 PRINT "Lion", "Whale", "Monkey", "Fish	Lion Whale Monkey Fish

b) INPUT statement:

INPUT statement is used to take input from the user during the execution of the program.

Syntax:

Line number INPUT [Sequence of characters/string;] List of variables

Sequence of characters/string is used to prompt the user to enter the required value using keyboard and it is optional. List of variables are used to store the entered values.

Line number INPUT [sequence of characters / string;] List of variables.

Example 1: Program to find the square of a number.

Programme:	Clear screen. Ask for input from the user. Print the string and square of entered number
10 CLS 20 INPUT X 30 PRINT "SQUARE of Your Number is = "; X^2	
RUN Enter Number - 5 SQUARE of Your Number is = 25	Prompt for user Input from key board Result is displayed on the screen

Example 2: Program to find the area of a circle.

Program	
10 CLS	Clear screem
20 PIE = 3.14	Assigns the value to the variable
30 INPUT "Enter Radius"; R	Ask the user to enter radius
40 A = PIE * R^2	Calculates the Area and assign it to variable A
50 PRINT "The Area of the Circle =";A	Prints the string and the variable
Run	
Enter Radius? 7.4	Prompt for user and Input from keyboard
The Area of the Circle is = 171.9464	Result is displayed on the screen

c) READ and DATA statements:

READ and DATA statements are used when there is a need to process large number of variables with given data. READ statement defines the list of variables while DATA statement contains constant values for the variables in READ statements. Values in READ and DATA statements should be separated by commas.

Syntax:

Line number READ List of Variables separated by commas.

Line number DATA List of Constants separated by commas.

Example: 10 READ X, Y, Z, K
20 DATA 8, 9, 13, 15

When the above program is executed the line number 10 with the READ statement followed by the variables X,Y,Z,K will assign constant values from the DATA statement in line number 20 in the same order. A one-to-one correspondence exists in READ-DATA statements, i.e. X=8, Y=9, Z=13, K=15.

If values in the DATA statements are

more than the variables in the READ statement then the extra values of DATA are ignored. But if the variables in READ statement are more than the values in the DATA statement then a syntax error encounters i.e. "Out of Data".

Example:

Program	
10 CLS	Clear screen
20 DATA 6, 15, 23, 44	Data Statement with values
30 READ W, Y, X, Z	Reads four variables
40 SUM1 = W + Y + X + Z	Finds the sum numbers in the variable SUM1
40 PRINT "Sum =", SUM 1	Prints the values of variables on the screen
RUN	
Sum =88	Output

d) IF-THEN-ELSE statement:

IF...THEN is a decision making statement, depending upon the decision, it takes some action or changes the order of execution. It helps the computer to check whether a relation is TRUE or FALSE.

Syntax:

IF relational expression **THEN** statement(s)1 **ELSE** statement(s)2

If the relational expression is true then statement(s)1 will be executed otherwise the statement(s)2 will be executed.

Example:

Program	
10 CLS	Clear screen
20 INPUT "ENTER FIRST VALUE", A	Ask input for first value
30 INPUT "ENTER 2ND VALUE", B	Ask input for second value
40 IF A<B THEN SMALL = A ELSE SMALL = B	Compares the two values and stores the smaller number in the variable SMALL
50 PRINT" Smaller number = "; SMALL	Prints the string and the smaller number
RUN	
ENTER FIRST VALUE 7	Ask for input for first value
ENTER 2ND VALUE 3	Ask for input for second value
Smaller number = 3	Output

Objective Type Questions

MULTIPLE CHOICE QUESTIONS

✲ **Choose the correct answer.**

1. A series of instructions for the computer to follow is called:
(a) Programming (b) Program
(c) Programming Language
(d) Variables

2. Computers do not understand:
(a) Human Language
(b) Machine Language
(c) Assembly Language
(d) Constants

3. Programming languages are used to prepare computer:
(a) Programming (b) Program
(c) Programming Language
(d) Variables

4. Programming languages are divided into two groups: high level languages and _____ languages.
(a) Uper level (b) Medium level
(c) a+b both (d) Low level

5. A programming language that allows the user to write instructions using English is called:
(a) Human Language
(b) Machine Language
(c) Assembly Language
(d) High Level Language

6. A programming language that is nearer to computer hardware is called:
(a) Human Language
(b) Low Level Language
(c) Assembly Language
(d) High Level Language