GW-BASIC

Computer Science 10th

V 2.1

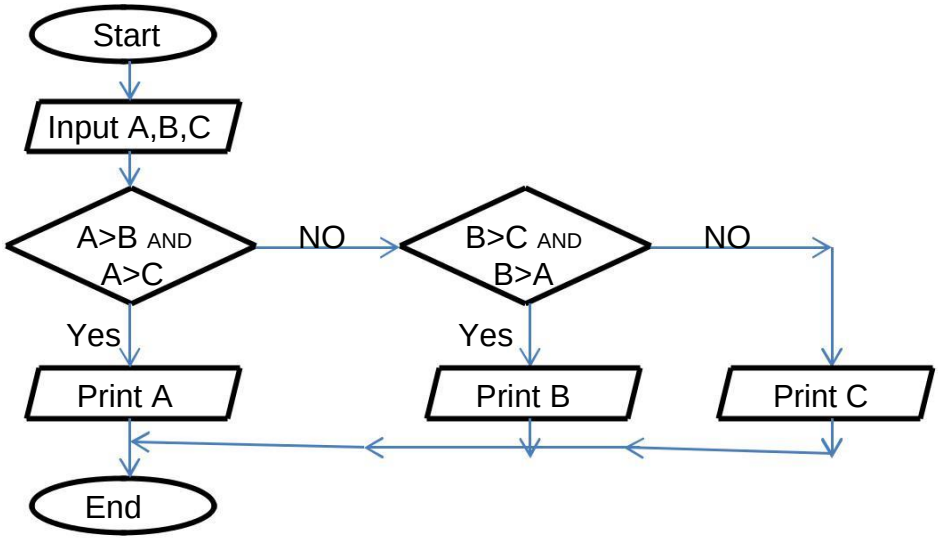
Exercise Programming Problems

Plus commands & functions

Complied By
M. Waqas Riaz
MS Computer Science
BS Electrical
(Telecommunication) Engineer
COMPUTERSCIENCECITY@GMAIL.COM
http://cscity.byethosy7.com

Chapter 1: Problem Solving

Question 8: Flow Chart to find greatest number among three numbers.



Question 9: Algorithm Area of Circle

SETP 1: Begin
STEP 2: Input Radius **R**
STEP 3: Area= 3.14 * R * R
STEP 4: Output Area
STEP 5: End

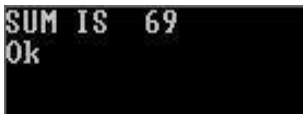
Question 10x:Algorithm covered distance

SETP 1: Begin
STEP 2: Input Average Speed **V**
STEP 3: Input Time **T**
STEP 4: Distance **S** = V * T
STEP 5: Output **S**
STEP 6: End

Chapter 2: Data Assignment & Input/output Statements

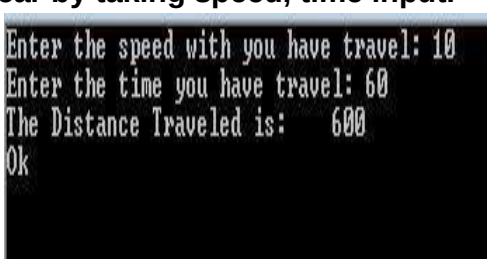
Q9: Program to read 10 values and find their sum.

```
10 CLS
20 READ A,B,C,D,E
30 READ F,G,H,I,J
40 READ I,J
50 SUM=A+B+C+D+E+F+G+H+I+J
60 PRINT "SUM IS ";SUM
70 DATA 5,2,1,8,10
80 DATA 20,3,7,9,4
90 END
```



Q10(x): Program to calculate distance travelled by a car by taking speed, time input.

```
10 CLS
20 INPUT "Enter the speed wich you have travel: ",V
30 INPUT "Enter the time you have travel: ",T
40 LET S=V*T
50 PRINT "The Distance Traveled is: ",S
60 END
```



Q10(ix):Program to calculate obtain marks & percentage by taking all subject's marks.

```
10  CLS
20  INPUT "Name: ",N$
30  INPUT "Roll No: ",ROLL
40  INPUT "Class: ",CLASS
50  INPUT "Section: ",SECTION$
60  PRINT "Enter the Marks  "
70  INPUT "English: ",ENG
80  INPUT "Urdu: ",URDU
90  INPUT "Islamiyat: ",ISL
100 INPUT "PAk Study: ",PST
110 INPUT "Maths: ",MATH
120 INPUT "Physics: ",PHY
130 INPUT "Chemistry: ",CHEM
140 INPUT "Computer: ",COMP
150 TOTA=850
```

```
Name: M Waqas Riaz
Roll No: 65
Class: 10
Section: B
Enter the Marks
English: 127
Urdu: 107
Islamiyat: 73
PAk Study: 67
Maths: 94
Physics: 96
Chemistry: 92
Computer: 91
Name: M Waqas Riaz          Roll No:  65  Class:  10 B
Obtain Marks:  747  Out Of:  850
Percentage:  87.88236
Ok
```

```
160 LET OBTAIN=ENG+URDU+ISL+PST+MATH+PHY+CHEM+COMP
170 LET PERCENTAGE=(OBTAIN/TOTAL)*100
180 PRINT "Name: ";N$   ,"Roll No: ";ROLL,"Class: ";CLASS;SECTION$
190 PRINT "Obtain Marks: ";OBTAIN;" Out Of: ";TOTAL
200 PRINT "Percentage: ";PERCENTAGE
210 END
```

Q10(xi): Program to demonstrate the difference between use of , & ; while printing.

```
10 CLS
20 A$="Punjab": B$="Sindh": C$="KPK": D$="Blochistan"
30 PRINT A$;B$;C$;D$
40 PRINT
50 PRINT A$,B$,C$,D$
60 END
```

```
PunjabSindhKPKBlochistan

Punjab          Sindh           KPK             Blochistan
Ok
```

Q11: Program to find the volume of the cylinder by inputting height and radius.

```
10 CLS
20 INPUT "Height of Cylinder: ", H
30 INPUT "Radius of Cylinder: ",R
40 LET V=3.14*R*R*H
50 PRINT "Volume Of Cylinder: ";V
60 END
```

```
Height of Cylinder: 5
Raduis of Cylinder: 2
Volume Of Cylinder:  62.80001
Ok
```

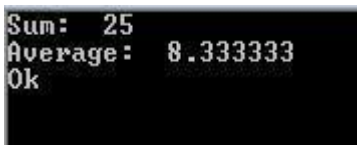
Q12: Program to find square of a number.

```
10 CLS
20 INPUT "enter the value ",X
30 SQ=X*X
40 PRINT "square is: ";SQ
50 END
```

```
enter the value 5
square is:  25
Ok
```

Q13: Program to calculate average of 3 numbers by LET statement.

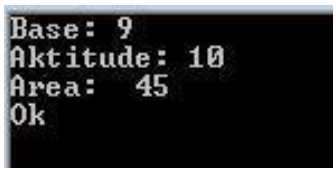
```
10 CLS
20 LET A=5
30 LET B=9
30 LET C=11
40 LET SUM=A+B+C
50 LET AVG=SUM/3
60 PRINT "Sum: ";SUM
70 PRINT "Average: ";AVG
80 END
```



Chapter 3: Control Structures

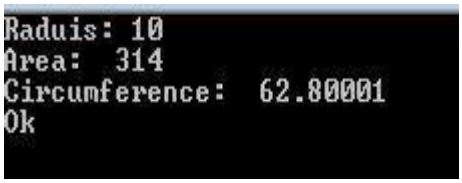
Q8: Program to calculate area of triangle by taking base and altitude as input.

```
10 CLS
20 INPUT "Base: ",B
30 INPUT "Altitude: ",A
40 AREA= (1/2)*B*A
50 PRINT "Area: ";AREA
60 END
```



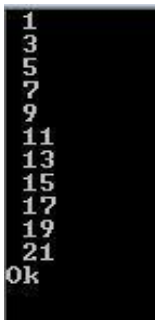
Q9: Program to calculate Circumference and area of a circle by taking radius input.

```
10 CLS
20 INPUT "Radius: ",R
30 A=3.14*R*R
40 C=2*3.14*R
50 PRINT "Area: ";A
60 PRINT "Circumference: ";C
70 END
```



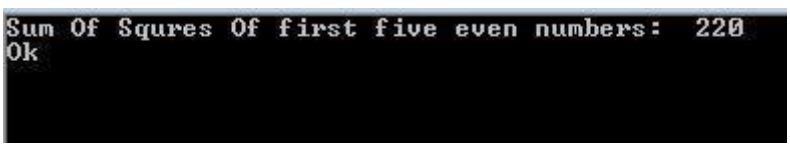
Q10: Program to print first ten odd numbers by while loop.

```
10 CLS
20 N=1
30 WHILE N<=21
40 PRINT N
50 N=N+2
60 WEND
70 END
```



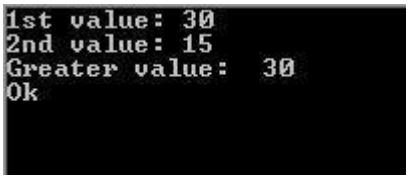
Q11: Program to calculate sum of squares of first five even numbers.

```
10 CLS
20 FOR N=2 TO 10  STEP 2
30 SUM= SUM+(N*N)
40 NEXT N
50 PRINT "Sum Of Squres Of first five even numbers: ";SUM
60 END
```



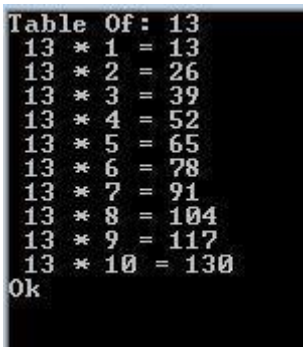
Q12: Program to find greater number among two values.

```
10 CLS
20 INPUT "1st value: ",A
30 INPUT "2nd value: ",B
40 PRINT "Greater value: ";
50 IF A>B THEN PRINT A ELSE PRINT B
60 END
```



Q13: Program to print table of a given number taken by input statement.

```
10 CLS
20 INPUT "Table Of: ",X
30 FOR N=1 TO 10
40 PRINT X;"*";N;"=";X*N
50 NEXT N
60 END
```



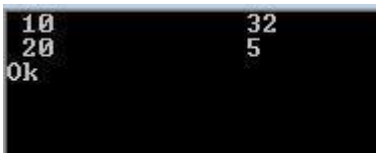
Q14: Program to calculate percentage and grade from obtain marks.

```
10  CLS
20  INPUT "Obtain Marks: ",OBT
30  P=(OBT/850)*100
40  IF P>=80 THEN PRINT "A1"
50  IF P>=70 AND P<80 THEN PRINT "A"
60  IF P>=60 AND P<70 THEN PRINT "B"
70  IF P>=50 AND P<60 THEN PRINT "C"
80  IF P>=40 AND P<50 THEN PRINT "D"
90  IF P<40 THEN PRINT "F"
100 END
```

Chapter 4: Arrays

Q9: Program to print 2D array in matrix form.

```
10 CLS
20 DIM A(2,2)
30 REM filling
40 FOR R=1 TO 2
50 FOR C=1 TO 2
60 READ A(R,C)
70 NEXT C
80 NEXT R
90 DATA 10,32,20,5
100 REM printing
110 FOR R=1 TO 2
120 FOR C=1 TO 2
130 PRINT A(R,C),
140 NEXT C
150 PRINT
160 NEXT R
170 END
```

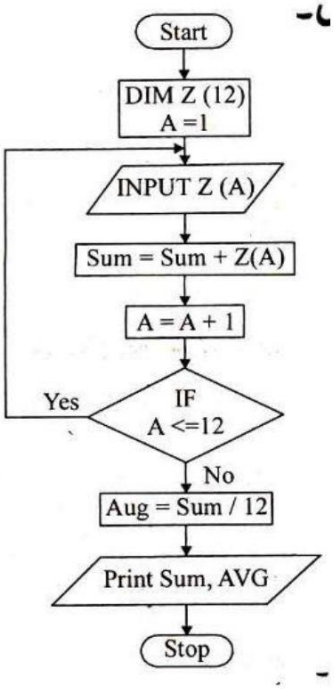


Q10: Program to print array in reverse order.

```
10  CLS
20  DIM A(10)
30  REM filling
40  FOR N=1 TO 10
50  INPUT "values of Array: ",A(N)
60  NEXT N
70  REM printing in reverse order
80  FOR N=10 TO 1  STEP -1
90  PRINT A(N)
100 NEXT N
110 END
```

```
values of Array: 9
values of Array: 2
values of Array: 5
values of Array: 9
values of Array: 10
values of Array: 3
values of Array: 4
values of Array: 5
values of Array: 98
values of Array: 30
 30
 98
 5
 4
 3
 10
 9
 5
 2
 9
Ok
```

Q12: Flow Chart for Q.16.



Q13: Algorithm to add 2 arrays A & B.

	کام
آغاز	عمل شروع کریں۔
پروسیسنگ	دو ارے کے ارکان کے لیے سٹوریج لوکیشنز محفوظ کریں۔
ان پٹ	پہلے ارے کے ارکان ارے A میں محفوظ کریں۔ دوسرے ارے کے ارکان ارے B میں محفوظ کریں۔
حسابی عمل	ارے A اور ارے B کے ارکان کو جمع کریں۔
آؤٹ پٹ	ارے A اور ارے B کے ارکان کو پرنٹ کریں۔
آؤٹ پٹ	ارے A اور ارے B کے حاصل جمع کو پرنٹ کریں۔
اختتام	عمل ختم کریں۔

STEP1: START
STEP2: CREATE 2 ARRAYS A & B
STEP3: INPUT ELEMENTS OF ARRAY A & B
STEP3: ADD ELEMENTS OF ARRAYS A & B
STEP4: PRINT ELEMENTS OF A & B
STEP5: PRINT ADDITION OF A &
B STEP6: END

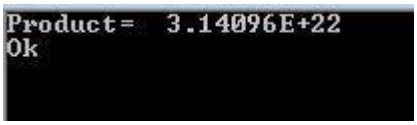
Q14: Program to print all even numbers from a given list.

```
10 CLS
02 DIM A(12)
02 FOR N=1 TO 12
02 READ A(N)
02 IF A(N) MOD 2 <> 0 THEN PRINT A(N)
02 NEXT N
02 DATA 6,42,4,77,39,9
02 DATA 21,22,8,45,15,46
02 END
```

```
 77
 39
 9
 21
 45
Ok
```

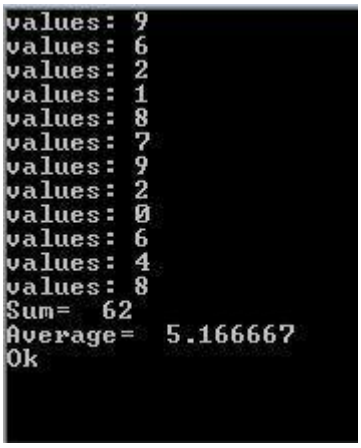
Q15: Program to find product of an array of 20 elements by reading values.

```
10 CLS
20 DIM A(20)
30 PRODUCT=1
40 FOR N=1 TO 20
50 READ A(N)
60 PRODUCT= PRODUCT*A(N)
70 NEXT N
80  PRINT "Product= ";PRODUCT
90  DATA 6,42,4,77,39,9,21,22,8,5
100 DATA 45,46,12,13,23,5,6,2,7,20
110 END
```



Q16: Program to calculate the sum & average of an array Z having 12 elements.

```
10  CLS
20  DIM Z(12)
30  SUM=0
40  FOR N=1 TO 12
50  INPUT "values: ",Z(N)
60  SUM=SUM+Z(N)
70  NEXT N
80  AVG=SUM/12
90  PRINT "Sum= ";SUM
100 PRINT "Average= ";AVG
110 END
```



Q18: Program to arrange 20 names in descending order.

```
10  CLS
20  DIM A$(20)
30  REM Filling
40  FOR I=1 TO 20
50  PRINT I;
60  INPUT "NAME: ", A$(I)
70  NEXT I
80  REM Arranging
90  FOR J=1 TO 20
100 FOR K=J+1 TO 20
110 IF A$(J)>A$(K) THEN 120 ELSE
150 120 TEMP$=A$(J)
130 A$(J)=A$(K)
140 A$(K)= TEMP$
150 NEXT K
160 NEXT J
170 REM PRINTING
180 FOR X=1 TO 20
190 PRINT A$(X)
200 NEXT X
210 END
```

Chapter 5: Subprograms and File Handling

Q11: Program to calculate the number of characters in first name of a person.

```
10  CLS
20  INPUT "Enter your Full Name(first<space>last): ",N$
30  CH$="A"
40  COUNT=0
50  WHILE CH$ <> " "
60  COUNT= COUNT+1
70  CH$= MID$(N$,COUNT,1)
80  WEND
90  LENGHT=COUNT-1
100 PRINT "Length Of first name is: "; LENGTH
110 END
```

```
Enter your Full Name(first<space>last): Muhammad Ali
Lenght Of first name is:  8
Ok
```

Q12: Program to print all the characters having ASCII values between 1 to 255.

```
10 CLS
20 FOR N=1 TO 255
30 PRINT CHR$(N)
40 NEXT N
50 END
```

Q13: Program to convert ^{O}C into ^{O}F by using DEF FN.

```
10 CLS
20 DEF FNF(C)= (9/5)*C +32
30 INPUT "Temperature in Celsius Scale: ",C
40 PRINT "Temperature in Fahrenheit: ",FNF(C)
50 END
```

```
Temperature in Celcius Scale: 37
Temperature in Fahrenheit:   98.6
Ok
```

Q14: Program to compute following formula.

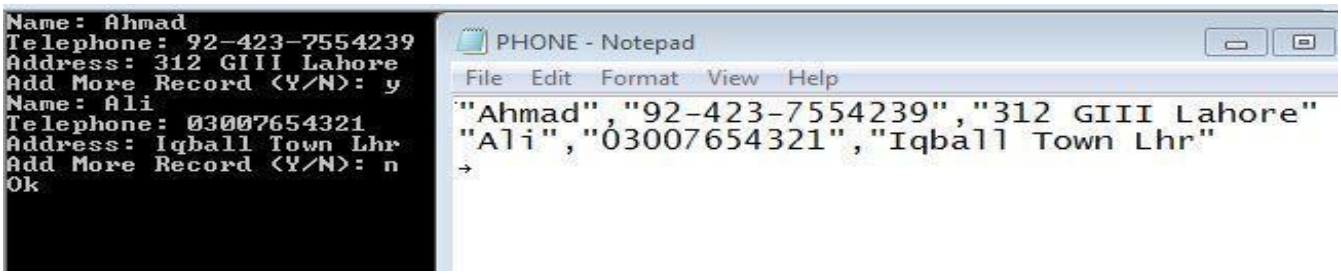
Combination = n! / k!(n-k)!

```
10 CLS
20 DEF FNC(N, K)= N!/K!*(N-K)!
30 INPUT "enter value of n: ",N
40 INPUT "Enter value of k: ",K
50 PRINT "Combination= ";FNC(N, K)
60 END
```

*factorial Part in function of This Program cannot be done by using concepts of 10th class, this this solution does not contain factorial part

Q15: Program to implement telephone directory which will be able to add contents.

```
10 CLS
20 OPEN "PHONE.DAT" FOR OUTPUT AS 1
30 INPUT "Name: ",N$
40 INPUT "Telephone: ",T$
50 INPUT "Address: ",ADD$
60  WRITE # 1,N$,T$,ADD$
70  INPUT "Add More Record (Y/N): ",A$
80  IF A$="Y" OR A$="y" THEN GOTO 30
90  CLOSE # 1
100 END
```



Chapter 6: Graphics In BASIC

Q9.a: Error

10 LINE (140,100)-(300-100),2,BF,4

Out Of Pixels

Q9.b: Error

10 **REM SCREEN 2 is black and white, no color is allowed correct is :**SCREEN 1 20 COLOR 1,2

30 DRAW "U10 R10 D10 L10"

Q9.c: Error

10 SCREEN 1

20 A=20

30 **REM DRAW "U=A R=A D=A L=A" CANT USE VARIABLE in draw, use constant:**

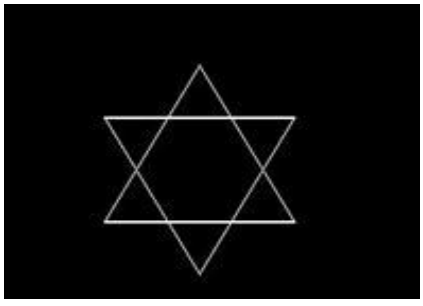
DRAW "U20 R20 D20 L20"

Q11: Program to draw a star.

```
10 CLS
20 SCREEN 2
30 PSET(250,50)
40 DRAW "G50 R100 H50"
50 PSET(250,120)
60 DRAW "E50 L100 F50"
70 END
```

OR

```
10 SCREEN 2
20 DRAW "G50 R100 H50 BD70 E50 L100 F50"
```

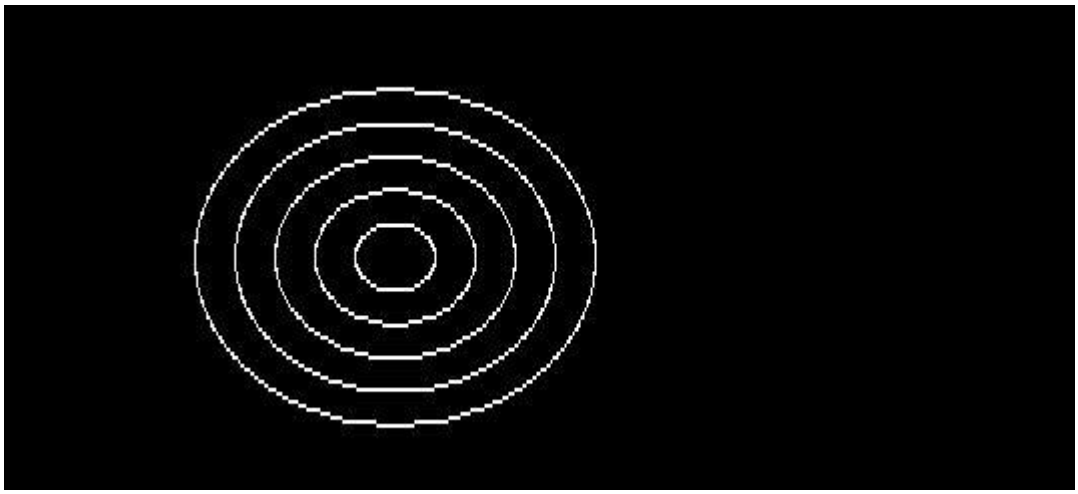


Q15: Program to draw 5 circles on same center.

```
10 CLS
20 SCREEN 2
30  CIRCLE(200,150), 30
 40 CIRCLE(200,150), 60
50  CIRCLE(200,150), 90
60  CIRCLE(200,150), 120
70  CIRCLE(200,150), 150
80 END
```

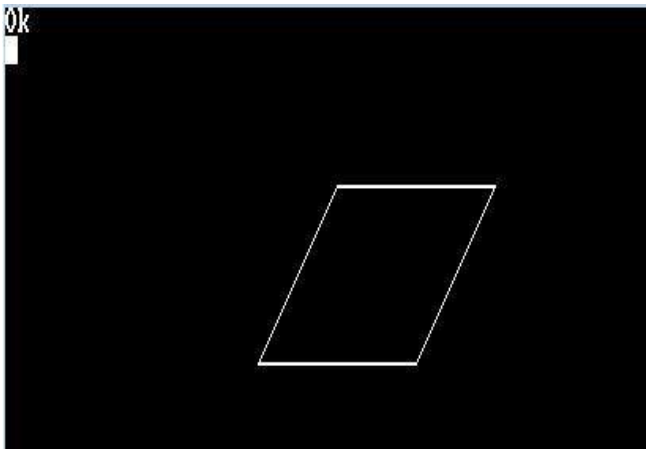
OR

```
10 CLS
20 SCREEN 2
30 FOR N=30 TO 150 STEP 30
40 CIRCLE(200,150), N
50 NEXT N
60 END
```



Q16: Program to draw a parallelogram.

```
10 CLS
20 screen 2
30 PSET(160,100)
40 DRAW "E50 R100 G50 L100"
50 END
```



SOME EXTRA PROGRAMS

Area of rectangle.

```
10    CLS
20    INPUT "Enter Length", L
30    INPUT "Enter Width", W
40    AREA= L*W
50    PRINT "Area=", AREA
60    END
```

Area and Volume of square.

```
10    CLS
20    INPUT "Enter the Length", L
30    AREA= L*L
40    VOL= L*L*L
50    PRINT "Area=", AREA
60    PRINT "Volume=", VOL
70    END
```

Convert ^{O}C into ^{O}F

```
10    CLS
20    INPUT "Temperature in Celsius
      Scale: ",C
30    F= (9/5)*C +32
40    PRINT "Temperature in
      Fahrenheit: ",F
50    END
```

Convert ^{O}C into ^{O}C

```
10    CLS
20    INPUT "Temperature in Fahrenheit
      Scale: ",F
30    C= 5*(F-32) /9
40    PRINT "Temperature in: Celsius", C
50    END
```

Greatest value among three numbers.

```
10    CLS
20    INPUT A,B,C
30    IF A>B AND A>C THEN 40
      ELSE 50
40    PRINT A;" is greatest" : END
50    IF B>A AND B>C THEN PRINT
      B;" is greatest": ELSE PRINT C;"
      is greatest"
60    END
```

Calculate answer according to option.
1-Addition 2-Subraction
3-Multiplication 4-Division

```
10    CLS
20    INPUT A,B
30    INPUT "Enter 1-Add 2-Subract 3-
      Multiply 4-Division", N
40    ON N GOTO 50,60,70,80
50    PRINT A+B : END
60    PRINT A-B : END
70    PRINT A*b : END
80    PRINT A/B : END
```

Find either given number is even/odd.

```
10    CLS
20    INPUT "Enter a Number", X
30    IF X MOD 2 = 0 PRINT "EVEN"
      ELSE PRINT "ODD"
40    END
```

Find either given number is +ve or -ve.

```
10    CLS
20    INPUT "Enter a Number", X
30    IF X >= 0 PRINT "positive" ELSE
      PRINT "negative"
40    END
```

Print the asterisk symbols * ^{to} form a triangular shape as follow by using loops.

```
10    CLS
20    FOR ROW=1 TO 5 STEP 1
30          FOR STAR= 1 TO ROW STEP 1
40          PRINT "*"
50          NEXT STAR
60    PRINT
70    NEXT ROW
80    END
```

```
*
**
***
****
*****
```

Print the asterisk symbols * to form a triangular shape as follow by using loops.

```
10    CLS
20    FOR ROW=5 TO 1 STEP -1
30        FOR STAR= 1 TO ROW STEP 1
40        PRINT "*"
50        NEXT STAR
60    PRINT
70    NEXT ROW
80    END
```

```
*****
****
***
**
*
```

Find the factorial of a given number.

```
10    CLS
20    INPUT "Enter a Number of which
      you want to calculate factorial", X
30    FACT=1
40    FOR I= 1 TO X STEP 1
50    FACT=FACT*I
60    NEXT I
70    PRINT "Factorial=",FACT
80    END
```

Find area of rectangle by DEF FN.

```
10    CLS
20    DEF FNA(L,W)= L*W
30    INPUT "Enter Length", L
40    INPUT "Enter Width", W
50    PRINT "Area=", FNA(L,W)
60    END
```

Print A to Z by using loop.

```
10    CLS
20    FOR I=65 TO 90 STEP 1
30    PRINT CHR$(I)
40    NEXT I
50    END
```

Draw an ellipse.

```
10    SCREEN 2
20    CIRCLE(100,100),50,,,,1
```

Draw 3 Circles of red color on same center.

```
10    SCREEN 7
20    CIRCLE (100,100),50,4
30    CIRCLE (100,100),70,4
40    CIRCLE (100,100),90,4
50    END
```

Draw a rectangle by LINE statements.

```
10    CLS
20    LINE (50,50)-(100,50)
30    LINE (100,50)-(100,100)
40    LINE (100,100)-(50,100)
50    LINE (50,100)-(50,50)
60    END
```

Draw a rectangle by DRAW statement.

```
10    SCREEN 2
20    DRAW "R100 D50 L100 U50"
```

Draw a Square by DRAW statement.

```
10     SCREEN  2
20     DRAW "R50 D50 L50 U50"
```

Draw a triangle by DRAW statement.

```
10    SCREEN 2
20    DRAW "G50 R100 H50"
```

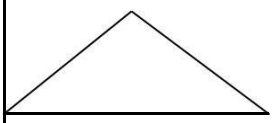
Draw a Square/BOX by LINE statement.

```
10     SCREEN
20     LINE (50,50)-(100,100),,B
```

GW-BASIC COMMANDS & FUNCTIONS

Sr.	COMMAND	PURPOSE	SYNTAX	EXAMPLES
01	**AUTO**	Generate the line numbers automatically.	AUTO [line no.]	AUTO AUTO 5 AUTO 100,5
02	**RENUM**	Renumber the program lines automatically.	RENUM [new no.]	RENUM RENUM 100
03	**LIST**	Display all or part of the program currently in memory on screen.	LIST [starting line no.]-[ending line no]	LIST LIST 10-100 LIST 100- LIST –100
04	**LLIST**	Print all or part of the program currently in memory from printer.	LLIST [starting line no.]-[ending line no]	LLIST 10-100
05	**DELETE**	Erase the specified range of lines of the program from computer memory.	DELETE [starting line no.]-[ending line no.]	DELETE 40 DELETE 50-200
06	**EDIT**	Display the specified line and change the line.	EIDT [line no.]	EIDT 50
07	**LOAD**	Reads a program from a specified device and store it in memory.	LOAD”file name.bas” [,r]	LOAD”TEST.BAS” LOAD”TEST.BAS””,R
08	**SAVE**	Save a program file in a disk.	SAVE”fileName.bas” SAVE”fileName.bas”[,a] SAVE”fileName.bas”[,p]	SAVE”TEST.BAS” SAVE”D:\TEST.BAS” SAVE”TEST.BAS”,A
09	**RUN**	Execute of the program currently in memory.	RUN [starting line no.]-[ending line no.]	RUN RUN 50-100
10	**NEW**	Delete the program currently in memory and clear all variables.	NEW	NEW
11	**SYSTEM**	Close all open files and return control (from GW-BASIC) to the operating system (WINDOWS).	SYSTEM	SYSTEM
12	**CLS**	CLS is used to clear the entire screen.	CLS [n]	CLS CLS 2
13	**REM**	To put a remarks or comments in a program.	REM [remarks] ‘ [remarks]	5 REM practical no.1 5 ‘ practical no.1
14	**INPUT**	To take the instruction from keyboard during the execution of program	INPUT “prompt string” [;/,] variable	INPUT “enter name”, N$
15	**PRINT**	To transmit and display the data and output on a screen.	PRINT “string” [;/,] [variable]	PRINT “YOUR NAME”,N$
16	**LPRINT**	To Prnt the data and output from the printer.	LPRINT “string” [;/,] [variable]	LPRINT “YOUR NAME”,N$
17	**LET**	To assign the value of expression to a variable.	LET variable = expression	LET A=4 LET N$=”ALI”
18	**GOTO**	To branches unconditionally to the specified line number.	GOTO line no.	100 GOTO 20

19	**KEY**	To assign the function key, display the list of function keys and on/off the key.	KEY n,"text", KEY LIST, KEY ON/OFF	KEY 2,"FILES" KEY LIST KEY ON KEY OFF
20	**FILES**	Display all names of files/programs from specified disk.	FILES [path]	FILES FILES D:\
21	**END**	To terminate the program execution and return to command level.	END	100 END
22	**STOP**	To terminate the program execution and return to command level.	STOP	100 STOP
23	**CONT**	Resume execution of an interrupted operation. Using Stop and End statement.	CONT	CONT
24	**KILL**	Delete a file/program from a diskette.	KILL "filename"	KILL "test.bas" KILL "New.txt"
25	**NAME**	To change the old file name in new file name.	NAME "old filename" AS "new filename"	NAME "test.bas" AS "Prog1.bas"
26	**SHELL**	To temporary exit from Basic to Dos command prompt.	SHELL	SHELL
27	**LOCATE**	Moves the cursor to the specified position.	LOCATE [row,column]	30 LOCATE 5,10
28	**SWAP**	To interchange the value of two variables.	SWAP variable, Variable	SWAP A, B SWAP A$,B$
29	**IF-THEN-ELSE**	To make a decision on the result of an expression.	IF [expression] THEN [statement] ELSE [statement]	IF a=10 THEN 70 ELSE 20 IF a>b THEN PRINT a ELSE PRINT B
30	**FOR-NEXT**	To perform a repetitive loop for given number of time.	FOR variable=starting no. TO ending no [STEP] Statements NEXT	10 FOR A=1 to 10 20 PRINT A 30 NEXT A
31	**GOSUB-RETURN**	To branch a subroutine and return from a subroutine.	GOSUB [line no.] Statements RETURN	30 GOSUB 40 40 PRINT "TEST SUB" 70 RETURN
32	**READ-DATA**	To assign the numeric or string values in variables.	READ variables DATA data	40 READ A,N$ 80 DATA 123,ALI
33	**DIM**	To assign the values of array variable subscript & allocate storage accordingly.	DIM variable(subscript)	DIM A (3), N$(3)
34	**WHILE-WEND**	To perform a conditional loop.	WHILE condition Statements WEND	10 A=1 20 WHILE A<10 30 PRINT A 40 A=A+1 50 WEND
35	**ON-GOTO**	Branches/Transfers to one or more specified line nos. depending on the value of expression.	ON [variable] GOTO [line no. list]	20 ON choice GOSUB 30,40,50
36	**CHAIN**	To call a program into memory and execute the program. [LOAD+RUN]	CHAIN "file name"	CHAIN "test.bas"

<table>
<tr><td>37</td><td><u>MERGE</u></td><td>To combine a program into the program currently in memory.</td><td>MERGE “file name”</td><td>MERGE “TEST.BAS”</td></tr>
<tr><td>38</td><td><u>MKDIR</u></td><td>To create a directory on the specified disk.</td><td>MKDIR“directory name”</td><td>MKDIR “abc”</td></tr>
<tr><td>39</td><td><u>CHDIR</u></td><td>To allow you to change the directory.</td><td>CHDIR “directory name”</td><td>CHDIR “abc”</td></tr>
<tr><td>40</td><td><u>RMDIR</u></td><td>To remove a directory from specified disk.</td><td>RMDIR “directory name”</td><td>RMDIR “abc”</td></tr>
<tr><td>41</td><td><u>OPEN</u></td><td>To open a file (Load into memory) for file handling.</td><td>OPEN “name” for mode as buffer</td><td>OPEN“PHONE.DAT” FOR INPUT AS 1</td></tr>
<tr><td>42</td><td><u>WRITE#</u></td><td>To send/write/save data in open file in memory.</td><td>WRITE # buffer, value</td><td>WRITE # 1, “hello”</td></tr>
<tr><td>43</td><td><u>INPUT#</u></td><td>To read (input) data (in program) from opened file</td><td>INPUT #buffer,variable</td><td>INPUT</td></tr>
<tr><td>44</td><td><u>EOF()</u></td><td>To test for end of file.</td><td>EOF(buffer)</td><td>IF EOF(1) THEN END</td></tr>
<tr><td>45</td><td><u>CLOSE</u></td><td>To close opened files.</td><td>CLOSE buffers</td><td>CLOSE 1,2,3</td></tr>
<tr><td>46</td><td><u>CLEAR</u></td><td>To clear memory (close files, finish variable values)</td><td>CLEAR</td><td>CLEAR</td></tr>
<tr><td colspan="5"><u>(GRAPHIC STATEMENTS)</u></td></tr>
<tr><td>47</td><td><u>SCREEN</u></td><td>To sets the specifications for the display screen</td><td>SCREEN [mode]</td><td>SCREEN 2</td></tr>
<tr><td>48</td><td><u>CIRCLE</u></td><td>To draw a circle, arc and ellipse on the screen.</td><td>CIRCLE(x,y),radius,[color]</td><td>CIRCLE (100,160), 80</td></tr>
<tr><td>49</td><td><u>LINE</u></td><td>To draw a line or box on the screen. Also used to draw a Box.</td><td>LINE (x1,y1)-(x2,y2),[color],[box]</td><td>LINE (10,50)-(30,40),2</td></tr>
<tr><td>50</td><td><u>DRAW</u></td><td>To draws an object as specified by a string expression
<table>
<tr><td>Dn</td><td>Move down</td><td>Fn</td><td>diagonally down right</td></tr>
<tr><td>Un</td><td>Move up</td><td>Gn</td><td>diagonally down left</td></tr>
<tr><td>Ln</td><td>Move left</td><td>Hn</td><td>diagonally up left</td></tr>
<tr><td>Rn</td><td>Move right</td><td>En</td><td>diagonally up right</td></tr>
<tr><td>Bn</td><td>Mover but no plot draw</td><td>Nn</td><td>Move but return to previous position</td></tr>
</table>
n=Number</td><td>DRAW “string expression”</td><td>DRAW “F60 L120 E60”
  </td></tr>
<tr><td>51</td><td><u>PSET</u></td><td>To set the pixel position on the screen.</td><td>PSET (x,y),[color]</td><td>PSET (50,80),2</td></tr>
<tr><td>52</td><td><u>PAINT</u></td><td>To fill a diagram or area with a color and pattern.</td><td>PAINT (x,y), [color, border, background]</td><td>PAINT (100,160),2</td></tr>
<tr><td colspan="5"><u>(FUNCTIONS)</u></td></tr>
<tr><td>53</td><td><u>ABS()</u></td><td>To return the absolute value of the numeric expression.</td><td>ABS(value/variable)</td><td>PRINT ABS(-5)
ANSWER: 5</td></tr>
</table>

54	**INT()**	To return the integer value.	INT(value)	PRINT INT(-50.45) Output: 50
55	**FIX()**	To return the value without fraction.	FIX(value)	PRINT FIX(-50.45) Output: -50
56	**ASC()**	To display the ASCII code for the character of the string.	ASC (Character)	PRINT ASC(“a”) Output: 65
57	**CHR$()**	To converts an ASCII code to its character equivalent.	CHR$ (value/variable)	PRINT CHR$(65) Output: a
58	**VAL()**	To convert the string value in numeric value.	VAL(string)	VAL(“23 D Block”)
59	**DATE$**	To display and set the system date.	DATE$	PRINT DATE$ DATE$=”10-11-2013”
60	**TIME$**	To display and set the system time.	TIME$	PRINT TIME$ TIME$=”12:30”
61	**LEFT$()**	To return the given number of character form left side.	LEFT$(string,n)	? LEFT$(“paksitan”,3 Output: pak
62	**RIGHT$()**	To return the given number of character from right side.	RIGHT$(string,n)	PRINT RIGHT$(“Pakistan”,2) Output: an
63	**MID$()**	To return the requested part of a given string.	MID$(string,n1,n2)	? MID$(“Pakistan”,4,2) Output: is
64	**LEN():**	To return the number of character in a string.	LEN(string)	? LEN(“Pakistan”) Output: 8
65	**SPACE$()**	To returns a sequence if blank spaces.	SPACE$(x)	SPACE$(8)
66	**SPC()**	To generate blank spaces in a print statement.	SPC(x)	PRINT SPC(5)
66	**TAB()**	To set the column position on the screen or printer.	TAB(x)	TAB(10)
67	**EXP()**	To calculate the exponential function. ***(e^x)***	EXP(value)	PRINT EXP(1) Output: 2.71***(e^1)***
68	**LOG()**	To return the natural logarithm of a value. ***(log_e)***	LOG(x)	LOG(2.5)
69	**SQR()**	To return the square root of a given value.	SQR(x)	PRINT SQR(64) Output: 8
70	**SIN()**	To calculate the trigonometric sine function.	SIN(x) x is angle in radians	SIN(30*3.14/180)
71	**COS()**	To calculate the trigonometric cosine function.	COS(x) x is angle in radians	COS(30*3.14/180)
72	**TAN()**	To return the trigonometric tangent function.	TAN(x) x is angle in radians	TAN(30*3.14/180)
73	**HEX$**	To convert the decimal value in hexadecimal value.	HEX$(x)	PRINT HEX$(47) Output: 2F
74	**OCT$()**	To converts the decimal value in octal value.	OCT$(x)	OCT$(45)
75	**INKEY$**	To read a single character from the keyboard.	INKEY$	Y$=INKEY$
76	**INPUT$()**	To read a characters from the keyboard.	INPUT$(n)	Y$= INPUT$(1)

From Where You Can Download GW-BASIC ?

You Can Download GW-BASIC Interpreter from CSCITY Website

http://cscity.byethost7.com
http://cscity.byethost7.com/?page_id=19 (X Class)

Click GWBASIC with Programs.zip To Download GW-BASIC Software. This Package Contained All Solved Program's Source Codes from Exercise of **Computer Science for Class 10th Punjab Text Book Board Lahore**. When You Will Install GW-BASIC From GWBASIC with Programs.zip All Programs Codes will Automatically Copied in Your GW-BASIC.

How To Install GW-BASIC from GWBASIC with Programs.zip ?

Simply Download GWBASIC with Programs.zip File From Website Open the File And Double Click On GWBASIC 32 n 64 bit with all programs(PTB 10th exercises).exe And Press Install Button, GW-BASIC Will Automatically Installed in You C: Drive. If You Did Not Install It In C: Drive, The 64bit Version Of GW-BASIC Will Not Work. This Setup Contain Both 32bit & 64bit Both Files, This Setup Can Be Run On All Windows Versions Including Windows 98, Windows 2000, Windows xp, Windows Vista, Windows 7 and Windows 8 both 32bit and 64bit versions.

There Are Already All Solutions Of All Programs Of Your Exercise. Just Write The Name Of The Question And Chapter Number With LOAD Command. For Example For x Part Of Q10 from Chapter no. 2 type **LOAD "CH2QX.BAS"**And Press ENTER. Or Type **LOAD "CH3Q12.BAS"** For Question no. 12 of Chapter no. 3. You Can Also Use **DIR** Command To View All The List of Files(Programs) Available in GW-BASIC.

Use **LIST** Command To View Program's Lines Which is currently Loaded In GW-BASIC.

Students Can Also Download Softcopy Of These and other Computer Science Notes Of Class IX, X, XI, XII & SOURCE CODES Of THESE Programs Including Translators (GW-BASIC, TURBO C, VB etc) From COMPUTER SCIENCE CITY Website: http://CsCity.byethost7.com/

Your Feedbacks Will Be Appreciated.

M. Waqas Riaz
MS Computer Science
BS Electrical (Telecommunications) Engineer
COMPUTERSCIENCECITY@GMAIL.COM
http://facebook.com/waqass.riaz

☺ WISH YOU BEST OF LUCK ☺

HAND NOTES: