CH # 3 CONDITIONAL LOGIC

3.1 CONTROL STATEMENT

SHORT QUESTIONS

Q.1 Define Control Statement. (K.B)

Ans:

CONTROL STATEMENT

"Control Statement controls the flow of execution of a program" Sometimes we need to execute one set of instructions if a particular condition is **TRUE** and another set of instructions if the condition is **FALSE**. Moreover, sometimes we need to repeat a set of statements for a number of times. We can control the flow of program execution through control statements. There are three types of control statements in C language.

1) Sequential Control Statements
2) Selection Control Statements
3) Repetition Control Statements

Q.2 Name Types of Control Statement. (K.B)

Ans:

TYPES OF CONTROL STATEMENT

There are three types of control statements in C language.

1. Sequential Control Statements
2. Selection Control Statements
3. Repetition Control Statements

Sequential Control:

Sequential control is the default control structure in C language. According to the sequential control, all the statements are executed in the given sequence.

Selection Statements:

The statements which help us to decide which statements should be executed next, on the basis of conditions, are called selection statements.

Two types of selection statements are:

1. If statement
2. If-else statement

Repetition Control:

The control structure which keeps on repeating a statement or a set of statements upto a fixed number of time or until an associated condition remains true.

Two types of repetition statements are:

1. FOR statement
2. WHILE statement
3. DO-WHILE statement

MUTLIPLE CHOICE QUESTIONS

1. _____ is the default control structure: (K.B)
(A) Sequence (B) Selection (C) Repetition (D) None of these

2. In _____ control structure the instructions are executed according to ascending order. (K.B)
(A) Sequence (B) Selection (C) Repetition (D) None of these

3. _____ statement are executed on basis of condition. (K.B+U.B)
(A) Sequential (B) Selection (C) both (D) None of these

4 Condition is always written in _____. (K.B)
(A) Quotes " " (B) Parentheses () (C) Braces{} (D) None of these

5. A condition can be _____ expression. (K.B)
(A) Relational (B) Logical (C) Arithmetic (D) All of these

3.2 SELECTION STATEMENTS

3.2.1 IF STRUCTURE
3.2.2IF-ELSE STRUCTURE
3.2.3NESTED SELECTION STRUCTURES.

LONG QUESTIONS

1. Define Selection Statement. Explain IF Statement in detail with an example. (K.B+U.B)

Ans:

SELECTION STATEMENTS

The statements which help us to decide which statements should be executed next, on the basis of conditions, are called selection statements.

Two types of selection statements are:

1. if statement
2. if-else statement

If statement:

C language provides **if statement** in which we specify a condition, and associate a code to it. The code gets executed if the specified condition turns out to be true, otherwise the code does not get executed.

Structure of if statement:

if statement has the following structure in C language:

if (condition)
Associated Code

1. In the given structure, if is a keyword that is followed by a condition inside parentheses ( ).
2. A condition could be any valid expression including arithmetic expressions, relational expressions, logical expressions, or a combination of these.

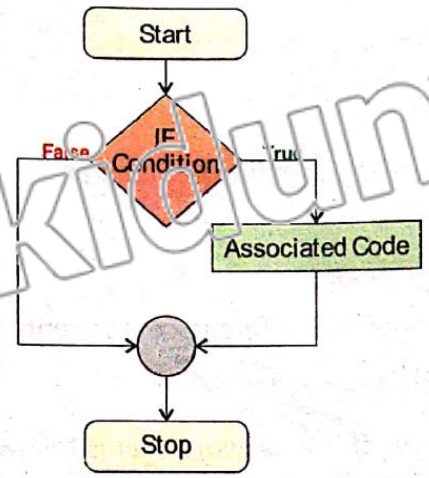
Example:

Here are a few examples of valid expressions that can be used as condition.

a- 5 (TRUE)
b- 5 + 4 (TRUE)
c- 5 – 5 (FALSE)

Any expression that has a non-zero value calculates to true, e.g. expressions a and b above produce a true value, but the expression c produces a false value. The expression can also include variables, in that case values inside the variables are used to calculate the true/false value of the expression.

3. The associated code is any valid C language set of statements. It may contain one or more statements. The following flow chart shows the basic flow of an if statement.



If we want to associate more than one statements to an **if statement**, then they need to be enclosed inside a **{ }** block, but if we want to associate only one statement, then although it may be enclosed inside **{ }** block, but it is not mandatory.

Example:

```
#include < stdio.h>
void main ( )
{
int a = 12;
if (a % 2 == 0)
{
printf (“The variable a contains an even value.”);
printf (“\nYou are doing a great job.”);
}
}
```

Output:

The variable a contains an even value.
You are doing a great job.

2. Define Selection Statement. Explain IF-ELSE Statement in detail with an example.(K.B+U.B)

Ans:

SELECTION STATEMENTS

The statements which help us to decide which statements should be executed next, on the basis of conditions, are called selection statements.

Two types of selection statements are:

1) if statement
2) if-else statement

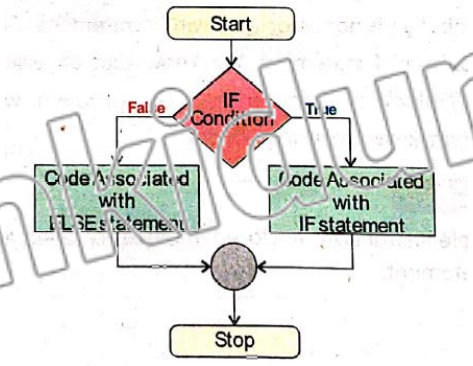
IF-ELSE STATEMENT

“**if-else** statement executes the set of statements under **if** statement if a condition is true and executes the set of statements under else otherwise”

General structure of the if-else statement is as follows:

```
if (condition)
Associated Code
else
Associated Code
```

Associated code of **if** statement is executed if the condition is TRUE, otherwise the code associated with **else** statement is executed. Following flow chart shows the structure of **if-else** statement.



An **if** statement may not have an associated **else** statement, but an **else** statement must have an **if** statement to which it is associated. Before **else** keyword, if there are multiple statements under if, then they must be enclosed inside the **{ }** block, otherwise compiler issues an error.

Example:

```
#include < stdio.h>
void main ( )
{
int a = 15;
if (a % 2 == 0)
{
printf ("The variable a contains an even value.");
printf ("\n You are doing a great job.");
}
else
printf ("The variable a contains an odd value.");
}
```

Output:

The variable a contains an odd value.

3. Define nested selection structure. Explain it with suitable examples. (K.B+U.B)

Ans Conditional statements within conditional statements are called nested selection structure. All the following structures are valid nested selection structures.

if (condition1 is true)
  if (condition2 is true)
  Associated code
  else
  Associated code
	if (condition1 is true)
	  if (condition2 is true)
	  Associated code
	  else
	  if (condition3 is true)
	  Associated code

if (condition1 is true)
  if (condition2 is true)
  Associated code
  else
  Associated code
  else
  if (condition3 is true)
  Associated code
	if (condition1 is true)
	  if (condition2 is true)
	  Associated code
	  else
	  Associated code
	  else
	  if (condition3 is true)
	  Associated code
	  else
	  Asscoted code

Example:

An electricity billing company calculates the electricity bill according to the following formula

Bill Amount = Number of Units Consumed **X** Unit Price

There are two types of electricity users i.e. Commercial and Home Users. For home users the unit price varies according to the following:

Units Consumed	Unit Price
Units < = 200	Rs 12
Units > 200 but Units < = 400	Rs 15
Units > 400	Rs 20

For commercial users, the unit price varies according to the following:

Units Consumed	Unit Price
Units < = 200	Rs 15
Units > 200 but Units < = 400	Rs 20
Units > 400	Rs 24

Write a program that takes the type of consumer and number of units consumed as input. The program then displays the electricity bill of the user.

Program:

```
#include<stdio.h>
void main()
{
        int units, unit_price, bill;
        char user_type;
        printf("Please enter h for home user and c for commercial user:  ");
        scanf("%c", &user_type);
        printf("Please enter the number of  units consumed:  ");
        scanf("%d", &units);
        if(units < = 200)
                if(user_type == 'h')
                        unit_price = 12;
                else if(user_type == 'c')
                        unit_price = 15;
        else if(units > 200 && units <= 400)
                if(user_type == 'h')
                        unit_price = 15;
                else if(user_type  ==  'c')
                        unit_price = 20;
        else
                if(user_type == 'h')
                        unit_price = 15;
                else if(user_type == 'c')
                        unit_price = 24;
        bill = units * unit_price;
        printf("Your electricity bill is %d",  bill);
}
```

The code associated with an ***if statement*** or with an ***else*** statement can be any valid 'C' language set of statements. It means that inside an ***if*** block or inside an ***else*** block, we can have other ***if*** statements or ***if-else*** statements. It also means that inside those inner ***if*** statement or ***if-else*** statements we can have even more ***if*** statements or ***if-else*** statements and so on.

SHORT QUESTIONS

Q.1 What are selection statements? (K.B)

Ans:

SELECTION STATEMENTS

The statements which help us to decide which statements should be executed next, on the basis of conditions, are called selection statements. Two types of selection statements are:

1. if statement 2. if-else statement

Q.2 Write the name of types of selection statements. (K.B)

Ans:

TYPES SELECTION STATEMENTS

Two types of selection statements are:

1. if statement 2.if-else statement

Q.3 Define Condition. (K.B)

Ans: A condition could be any valid expression including the arithmetic expressions, relational expressions, logical expression, or a combination of these. Condition always evaluates in TRUE or FALSE.

Q.4 Define IF statement. (K.B)

Ans:

IF STATEMENT

"if-else statement executes the set of statements under if statement if a condition is TRUE and executes the set of statements under else otherwise"

Syntax:

if (condition)
Associated Code

Q.5 Give Syntax of IF Statement. (K.B+U.B+A.B)

Ans:

SYNTAX OF IF STATEMENT

if statement has the following structure in C language:

if (condition)
Associated Code

Q.6 What do you know about associated code? (K.B+U.B)

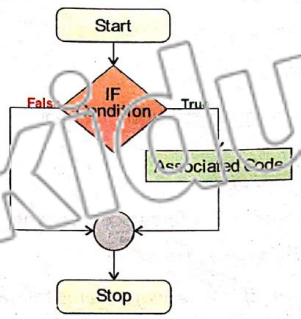
Ans:

ASSOCIATED CODE

The associated code is any valid C language set of statements. It may contain one or more statements.

Q.7 Make a Flow chart to explain IF structure. (K.B+U.B)

Ans:



Q.8 Define IF-ELSE statement. (K.B)

Ans:

IF-ELSE STATEMENT

"**if-else** statement executes the set of statements under **if** statement if a condition is TRUE and executes the set of statements under else otherwise"

Q.9 Give syntax of IF-ESLE Statement (K.B+U.B)

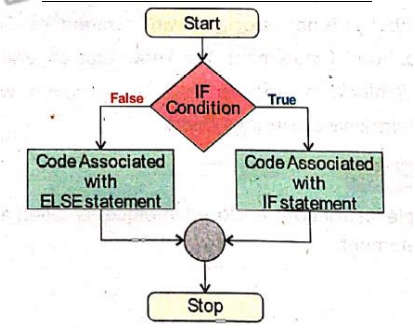
Ans: **IF-ESLE STATEMENT**

General structure of the if-else statement is as follows:

```
if (condition)
Associated Code
else
Associated Code
```

Q.10 Make a flow chart to explain if-else structure. (K.B+U.B)

Ans: **IF-ELSE STRUCTURE**



Q.11 Define block. (K.B)

Ans: A set of multiple instructions enclosed in braces is called a block or compound statement.

Q.12 What will happen if the multiple statements of IF statement are not enclosed in { } before use of ELSE statement. Show with an example.

Ans: Before else keyword, if there are multiple statements under if, then they must be enclosed inside the { } block, otherwise compiler issues an error. In order to understand this concept, let's look at the following example.

Example:

```
#include < stdio.h>
void main ( )
{
int a = 15;
if (a % 2 == 0)
{
printf ("The variable a contains and even value.");
printf ("/nYou are doing a great job.");
}
else
printf ("The variable a contains an odd value.");
}
```

Q.13 Define IF-ELSE-IF statement. (K.B)

Ans: **IF-ELSE-IF STATEMENT**

if-else-if statement also known as **else-if** statement. It is also a type of selection statement. It is used in a program when we have to use multiple conditions with an if statement.

Syntax:

```
if (condition 1)
        Code to execute if condition 1 is true;
else if (condition 2)
Code to execute if condition 1 is false but condition 2 is true;
```

.
.
.
.
.
else if (condition N)
code to execute if all previous conditions are false but condition N is true;
else
Code to execute if all the conditions are false;

Q.14 Give Syntax of IF-ELSE-IF Statement. (K.B+U.B)

Ans: **SYNTAX OF IF-ELSE-IF STATEMENT**

General syntax of the if-else-if statement is as follows:

```
if (condition 1)
Code to execute if condition 1 is TRUE;
else if (condition 2)
Code to execute if condition 1 is FALSE but condition 2 is TRUE;
.
.
.
.
.
else if (condition N)
code to execute if all previous conditions are FALSE but condition N is TRUE;
else
Code to execute if all the conditions are FALSE;
```

Q.15 Define Nested Selection Structure. (K.B)

Ans: **SELECTION STRUCTURE**

Conditional statements within conditional statements are called nested selection structures.

Q.16 Write the syntax of nested selection structure. (K.B)

Ans: **SYNTAX OF NESTED SELECTION STRUCTURE**

All the following structures are valid nested selection structures.

if (condition1 is true)
  if (condition2 is true)
  Associated code
  else
  Associated code
	if (condition1 is true)
	  if (condition2 is true)
	  Associated code
	  else
	  if (condition3 is true)
	  Associated code

if (condition1 is true)
  if (condition2 is true)
  Associated code
  else
  Associated code
  else
  if (condition3 is true)
  Associated code
	if (condition1 is true)
	  if (condition2 is true)
	  Associated code
	  else
	  Associated code
	  else
	  if (condition3 is true)
	  Associated code
	  else
	  Asscoted code

MUTLIPLE CHOICE QUESTIONS

1. _____ is the default control structure: (K.B)

(A) Sequence (B) Selection (C) Repetition (D) None of these

2. In _____ control structure the instructions are executed according to ascending order. (K.B+U.B)

(A) Sequence (B) Selection (C) Repetition (D) None of these

3. _____ statement are excited on basis of condition. (K.B+U.B)

(A) Sequential (B) Selection (C) Both A & B (D) None of these

4. Condition is always written in _____. (K.B)

(A) Quotes " " (B) parentheses (C) Braces {} (D) None of these

5. A condition can be _____ expression. (K.B)

(A) Relational (B) Logical (C) Arithmetic (D) All of these

6. True is indicated by: (K.B)

(A) 1 (B) 0 (C) Both A & B (D) None of these

7. False is indicated by: (K.B)

(A) 0 (B) 1 (C) Both A & B (D) None of these

8. _____ is a valid C language set of statement/s. (K.B)

(A) Associated Code (B) Condition (C) Expression (D) None of these

9. If we want to more than one statement with if statement then they needs to be enclosed with _____. (K.B+U.B)

(A) {} (B) ( ) (C) " " (D) None of these

10. Using _____ can improve the readability of program. (K.B)

(A) Spaces (B) Tag (C) Enter (D) None of these

11. If we don't use {} before Else statement if we have multiple statements, then _____.

(A) Complier issue an error (B) Complier ignore it

(C) Complier ignores else (D) None of these

12. Set of multiple statement enclosed in braces is called _____. (K.B)

(A) Multiple (B) Compound Statement (C) Group (D) None of these

13. A _____ is a group of statement enclosed in {} called _____ (K.B)

(A) Block (B) Group (C) Multiple (D) All of these

14. Code to execute after else statement in If -else- if statement only when. (K.B+U.B)

(A) 1st condition is true (B) 2nd condition is true

(C) 3rd condition is true (D) 4th condition is true

15. It is common mistake to omit_____. (K.B+U.B+A.B)

(A) keywords (B) Braces (C) Parenthesis (D) None of these

16. _____ is applicable in only limited scenarios. (K.B)

(A) if-else (B) if (C) if-else-if (D) switch –case

PROGRAMMING TIME (A.B)

Programming Time 3.1

Problem:
Write a program in C language that takes the percentage of student as an input and displays "PASS" if the percentage is above 50.

```
#include  <stdio.h>
void main()
{
        float percentage;
        printf ("Enter the percentage: ");
        scanf  ("%f, &percentage);
        if (percentage  >  50)
                printf ("PASS\n");
}
```

Output:
On the input 47, program simply ends because 47 is less than 50 and the condition turns false.

```
Enter the percentage: 47
```

When 67.3 is entered as an input, "PASS" gets printed on console because condition is true, as 67.3 is greater than 50.

```
Enter the percentage: 67.3
PASS
```

Programming Time 3.2

Problem:
A marketing firm calculates the salary of its employees according to the following formula.

Gross Salary = Basic Salary + (Number of Items Sold X8) + Bonus

If the number of sold items are more than 100 and the number of broken items are 0, then bonus is Rs. 10000, otherwise bonus is 0.
Write a program that takes basic salary, the number of sold and broken items as input from user, then calculates and displays the gross salary of the employee.

Program:

```
#include<stdio.h>
void  main()
{
```

```
        int basic_salary, items_sold, items_broken, gross_salary;
        int bonus = 0;
        printf("Enter the basic salary: ");
        scanf("%d, &basic_salary);
        printf("Enter the number of items sold: ");
        scanf("%d", &items_sold);
        printf("Enter the number of items broken: ");
        scanf("%d, &items_broken);
        if (items_sold > 100 && items_broken == 0)
                bonus = 10000;
        gross_salary = basic_salary + (items_sold * 8) + bonus;
        printf("Gross salary of the employee is %d", gross_salary);
}
```

Description:

In the above example, bonus is initialized to 0 because if the number of sold items are not more than 100, then automatically bonus is considered 0. Inside the if statement, it is checked that whether the number of sold items are greater than 100. If so, the bonus is assigned 10000. It is to be noted that gross salary is calculated outside the if block, because whether the number of sold items are more than 100 or not, the gross salary must be calculated.

Programming Time 3.3

Problem:

Write a program that takes percentage marks of student as input and displays his grade. Following table shows grades distribution criteria.

Percentage	Grade
80% and above	A
70% – 80%	B
60% – 70%	C
50% – 60%	D
Below 50%	F

Program:

```
#include<stdio.h>
void main()
{
        float percentage;
        printf ("Enter the percentage:   ");
        scanf ("%f", &percentage);
        if (percentage >= 80)
                printf   ("A\n");
        else if (percentage   >= 70)
                printf   ("B\n");
        else if (percentage   >= 60)
                printf   ("C\n");
        else if (percentage   >= 50)
                printf   ("D\n");
        else
                printf   ("F\n");
}
```

Programming Time 3.4

Problem:

An electricity billing company calculates the electricity bill according to the following formula.

Bill Amount = Number of Units Consumed X Unit Price

There are two types of electricity users i.e. Commercial and Home Users. For home users the unit price varies according to the following:

Units Consumed	Unit Price
Units < = 200	Rs 12
Units > 200 but Units < = 400	Rs 15
Units > 400	Rs 20

For commercial users, the unit price varies according to the following:

Units Consumed	Unit Price
Units < = 200	Rs 15
Units > 200 but Units < = 400	Rs 20
Units > 400	Rs 24

Write a program that takes the type of consumer and number of units consumed as input. The program then displays the electricity bill of the user.

Program:

```
#include<stdio.h>
void main()
{
        int units, unit_price, bill;
        char user_type;
        printf("Please enter h for home user and c for commercial user:  ");
        scanf("%c", &user_type);
        printf("Please enter the number of  units consumed:  ");
        scanf("%d", &units);
        if(units < = 200)
                if(user_type == 'h')
                        unit_price = 12;
                else if(user_type == 'c')
                        unit_price = 15;
        else if(units > 200 && units <= 400)
                if(user_type == 'h')
                        unit_price = 15;
                else if(user_type  ==  'c')
                        unit_price = 20;
        else
                if(user_type == 'h')
                        unit_price = 15;
                else if(user_type  ==  'c')
                        unit_price = 24;
        bill = units * unit_price;

        printf("Your electricity bill is %d",  bill);
}
```

Programming Time 3.5

Program:

Write a program that displays larger one out of the three given number.

Program:

```
include <stdio.h>
void main()
{
        int n1, n2, n3;
        printf (“Enter three numbers”);
        scanf (“%d%d%d”, &n1, &n2, &n3);
        if (n1 > n2 &&n1 > n3)
                printf (“The largest number is %d”, n1);
        else if (n2  > n3 && n2  > n1)
                printf (“The largest  number is %d”, n2);
        else
        printf (“The largest  number is %d”, n3);
}
```

Programming Time 3.6

Problem:

Write a program that calculates the volume of cube, cylinder or sphere, according to the choice of user.

Program:

```
#include<stdio.h>
void main ()
{
        int choice;
        float volume ;
        printf (“Find Volume\n”);
        printf (“1.Cube\n2.Cylinder\n3.Sphere\nEnter your choice: ");
        scanf  (“%d”, &choice);
        if (choice == 1)
```

```
{
        float length;
        printf (“Enter Length: ”);
        scanf (“%f”, &length);
        volume = length * length * length;
        printf (“Volume is %f”, volume);
}
else if (choice == 2)
{
        float length1, radiusl;
        printf (“Enter Length: ”);
        scanf (“%f”, &length1);
        printf (“Enter Radius: ”);
        scanf (“%f”, &radius1);
        volume = 3.142 * radius1 * radius1 * length1;
        printf (“Volume is %f”, volume);
}
        else  if (choice == 3)
        {
                float radius;
                printf (“Enter Radius: ”);
                scanf (“%f”, &radius);
                volume = 3.142 * radius * radius * radius;
                printf (“Volume is %f”, volume);
        }
        else
                printf (“Invalid Choice”);
}
```

SOLVED ACTIVITIES (A.B)

ACTIVITY 3.1

Write a program that takes the age of a person as an input and displays "Teenager" if the age lies between 13 and 19.

Solution:

```
# include <stdio.h>
void main ()
{
int age;
printf("Enter your age");
scanf("%d",&age);
if(age>=13&&age<=19)
printf("Teenager");
else
printf("Not Teenager");
}
```

ACTIVITY 3.2

Write a program that takes year as input and displays "Leap Year" if the input year is leap year. Leap years are divisible by 4.

Solution

```
# include <stdio.h>
void main ()
{
int year;
printf("Enter a year");
scanf("%d",&year);
if(year%4==0)
printf("Leap Year");
else
printf("Not Leap Year");
}
```

ACTIVITY 3.3

Write a program that takes the value of body temperature of a person as an input and displays "You have fever." if body temperature is more than 98.6 otherwise displays "You don't have fever."

Solution

```
# include <stdio.h>
void main ( )
{
float  body_temp;
printf("Enter your body temperature");
scanf("%f",&body_temp);
if(body_temp>98.6)
printf("You have Fever");
else
printf("you don't have fever");
}
```

ACTIVITY 3.4

The eligibility criteria of university for its different undergraduate student programs is as follows:
BSSE Program : 80% or more marks in Intermediate
BSCS Program : 70% or more marks in Intermediate
BSIT Program : 60% or more marks in Intermediate
Otherwise the university do not enroll a student in any of its programs.
Write a program that takes the percentage of Intermediate marks and tells for which programs the student is eligible to apply.

Solution

```
# include <stdio.h>
void main ( )
{
float marks_percentage;
printf("Enter the percentage of intermediate marks");
scanf("%f",&marks_percentage);
if(marks_percentage>=80)
printf("You are eligible for BSSE program");
else if(marks_percentage>=70&&marks_percentage<80)
printf("you are eligible for BSCS program");
else if(marks_percentage>=60&&marks_percentage<70)
printf("You are eligible for BSIT");
else
printf("You are not Eligible for any program");
}
```

ACTIVITY 3.5

Write a program that takes two integers as input and asks the user to enter a choice from 1 to 4. The program should perform the according to the given table

CHOICE	OPERATION
1	Addition
2	Subtraction
3	Multiplication
4	Division

Solution

```
# include <stdio.h>
void main ( )
{
int num_1,num_2;
int choice;
printf("Enter first number");
scanf("%d",&num_1);
printf("Enter second number");
scanf("%d",&num_2);
printf("Enter your choice press 1 for Addition press 2 for Subtraction
press 3 for Multiplication press 4 for Division ");
if(choice==1)
printf("Addition=%d",num_1+num_2);
else if(choice==2)
printf("Subtraction=%d",num_1-num_2);
else if(choice==3)
printf("Multiplication=%d",num_1*num_2);
else if(choice==4)
printf("Division=%d",num_1/num_2);
else
printf("Invalid Input");
}
```

ACTIVITY 3.6

Write a program that finds and display s area of a triangle, parallelogram, rhombus or trapezium according to the choice of user.

Solution

```
# include <stdio.h>
void main ( )
{
int choice;
printf("Enter your choice to find the area of figure\n 1 for Triangle\n 2 for Parallelogram\n 3
                for Rhoumbus\n 4 for Trapezium");
scanf("%d", &choice);
if(choice==1)
{ //This block will calculate the area of triangle
float b,h,area;
printf("Enter the base & height of Triangle");
scanf("%f%f", &b,&h);
area=1/2*b*h;
printf("The area of figure=%f",area);
}
else if(choice==2)
{ //This block will calculate the area of Parallelogram
float b,h,area;
printf("Enter the base & height of Parallelogram");
scanf("%f%f", &b,&h);
area=b*h;
printf("The area of figure=%f",area);
}
else if(choice==3)
{ //This block will calculate the area of Rhoumbus
float b,h,area;
printf("Enter the base & height of Rhoumbus");
scanf("%f%f", &b,&h);
area=b*h;
printf("The area of figure=%f",area);
}
else if(choice==4)
{ //This block will calculate the area of Trapezium
float h,side1,side2,area;
printf("Enter the height,side1 & side 2 of Trapezium");
scanf("%f%f%f",&h,&side1, &side2);
area=1/2*(side1+side2)*h;
printf("The area of figure=%f",area);
}
else
printf("You have enter the wrong choice");
}
```

EXERCISE

Q1. Multiple Choice Questions.

1) Conditional logic helps in _____. (K.B)
(A) Decisions (B) Interactions (C) Traversing (D) All of these

2) _____ statements describe the sequence in which statements of the program should be executed. **(K.B)**
(A) Loop (B) Conditional (C) Control (D) All of these

3) In if statement, what happens of condition is false? (K.B)
(A) Program crashes (B) Index Out of Bound Error
(C) Further code executes (D) Compiler asks to change condition

4) (K.B)
```
int a = 5 ;
if (a < 10)
a++ ;
else
if (a > 4)
a --;
```
Which one of the following statements will execute?
(A) a++ (B) b--; (C) both (A) and (B) (D) None of these

5) Which of the following is the condition to check 'a' is a factor of 'c'? (K.B+U.B)
(A) a % c = = 0 (B) c % a = = 0 (C) a*c = = 0 (D) a + c = =0

6) A condition can be any _____ expression. (K.B+U.B)
(A) arithmetic (B) relational
(C) logical (D) arithmetic, relational or logical

7) An if statement inside another if statement is called _____ structure. (K.B)
(A) nested (B) boxed (C) repeated (D) decomposed

8) A set of multiple instruction enclosed in braces is called a _____. (K.B+U.B)
(A) box (B) list (C) block (D) job

ANSWER KEY

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

Q2. Define the following terms. (K.B)

1) Control Statements

Ans:

CONTROL STATEMENTS

"Control Statement controls the flow of execution of a program"

Sometimes we need to execute one set of instructions if a particular condition is true and another set of instructions if the condition is false. Moreover, sometimes we need to repeat a set of statements for a number of times. We can control the flow of program execution through control statements. There are three types of control statements in C language.

1. Sequential Control Statements
2. Selection Control Statements
3. Repetition Control Statements

2) Selection Statements

Ans:

SELECTION STATEMENTS

The statements which help us to decide which statements should be executed next, on the basis of conditions, are called selection statements.

Two types of selection statements are:

1. if statement
2. if-else statement

3) Sequential Statements

Ans: **SEQUENTIAL STATEMENTS**

Sequential control is the default control structure in C language. According to the sequential control, all the statements are executed in the given sequence.

4) Condition

Ans: A condition could be any valid expression including the arithmetic expressions, relational expressions, logical expression, or a combination of these. Condition always evaluates in true or false.

5) Nested Selection Structure

Ans: **NESTED SELECTION STRUCTURE**

A selection statement within another selection statements is known as Nested Selection Structure. The general structure of an if-else statement given below:

if (condition)
Associated Code
else
Associated Code

Q3. Briefly Answer the Following. **(K.B+U.B)**

1) Why do we need selection statements?

Ans: **SELECTION STATEMENTS**

The selection statements help us to decide which statements should be executed next, on the basis of conditions. These statements allows us to choose between the alternative program statements.

2) Differentiate between sequential statements and selection statements.

Ans: **STATEMENTS AND SELECTION STATEMENTS**

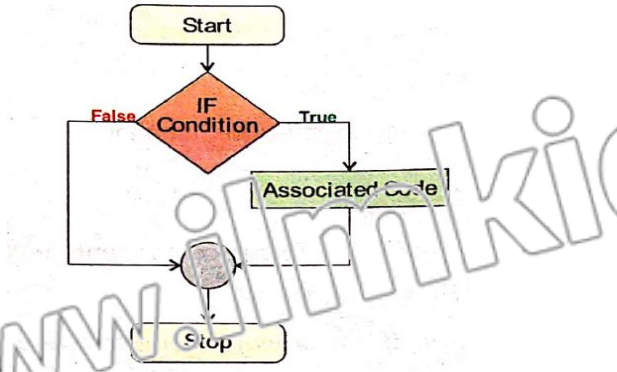
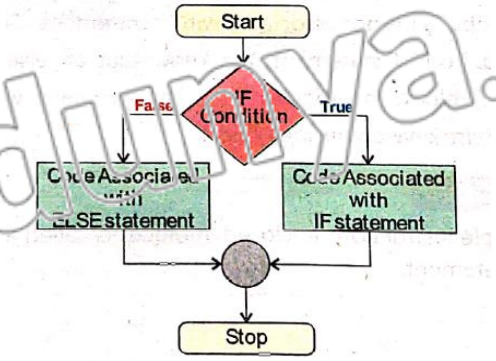
Sequential Statements	Selection Statements
Sequential control is the default control structure in C language. According to the sequential control, all the statements are executed in the given sequence.	The selection statements help us to decide which statements should be executed next, on the basis of conditions. These statements allows us to choose between the alternative program statements.

3) Differentiate between IF statements and IF-ELSE statement with an example.

Ans: **IF STATEMENTS AND IF ELSE STATEMENT**

IF STATEMENT	IF ELSE STATEMENT

if statement:
Definition
C language provides if statement in which we specify a condition, and associate a code to it. The code gets executed if the specified condition turns out to be TRUE, otherwise the code does not get executed.
Structure of if statement:
If statement has the following structure in C language:
If (condition)
Associated Code
In the given structure, if is a keyword that is followed by a condition inside parentheses ( ). A condition could be any valid expression including arithmetic expressions, relational expressions, logical expressions, or a combination of these.
	if-else Statement:
	Definition
	"if-else statement executes the set of statements under if statement if a condition is TRUE and executes the set of statements under else otherwise"
	Structure of if else statement:
	General structure of the if-else statement is as follows:
	if (condition)
	Associated Code
	else
	Associated Code
	Associated code of if statement is executed if the condition is TRUE, otherwise the code associated with else statement is executed. Following flow chart shows the structure of if-else statement.

Flowchart of if statement	Flowchart of if else statement
	

Example:
#include < stdio.h>
void main ( )
{
int a = 12;
if (a % 2 = = 0)
printf ("The variable a contains an even value.");
printf ("\nYou are doing a great job.");
}
}
	Example:
	#include < stdio.h>
	void main ( )
	{
	int a = 15;
	if (a % 2 == 0)
	{
	printf ("The variable a contains an even value.");
	printf ("/nYou are doing a great job.");
	}
	else
	printf ("The variable a contains an odd value.");
	}

4) What is the use of nested selection structures?

Ans: **SELECTION STRUCTURES**

Definition:

Conditional statements within conditional statements are called nested selection structures.

Format of Nested Selection Structure

if (condition 1 is TRUE)
if (condition 2 is TRUE)
associated code
else
associated code
else

5) Write the structure of if statement with brief description

Ans: **STRUCTURE OF IF STATEMENT**

If statement has the following structure in C language:

if (condition)
Associated Code

In the given structure, if is a keyword that is followed by a condition inside parentheses ( ). A condition could be any valid expression including arithmetic expressions, relational expressions, logical expressions, or a combination of these.

Q4. Identify the errors in following code segments. Assume that variables have already been declared. (K.B+U.B+A.B)

Ans:

Sr #	Program	Errors
(a)	if(x≥10) printf("Good");	Error expected in **if statement** we cannot write expression ≥ like that instead of that we have to write >=

(b)	if (a<b && b<C);
	sum = a+b+c;
	else
	multiply = a*b*c;
		Error expected in **if statement** we cannot use statement terminator; at the end of if statement.

(c)	if(a < 7 < (B) printf("7");	Error expected in **if statement** Logical operator (&& / ||) is not used in if statement.

(d)	if(a ==b &| x==y)
	flag = true;
	else
	flag = false;
		Error expected in **if statement** syntax of AND operator is not right logical operator cannot be write &| like that instead of that it can be write && like that.

e)	If(sum == 60 || product == 175)
	printf("Accepted %(C),sum);
	else
	if(sum>=45 || product > 100)
	printf("considered %d" + sum);
	else
	printf("Rejected");
		• Error expected in **line#2** (printf function) inverted comma's (string literals) are not used after format specifier.
		• Error expected in **line#5** (printf function) + sign is not allowed you have to use comma (,) at that place.

Q5. Write down output of the following code segments. (K.B+U.B+A.B)

Ans:

Sr #	Program	Output

(a)	int a = 7 , b = 10;
	a = a + b;
	if (a > 20 && b < 20)
	b = a + b ;
	printf ("a = %d , b = %d", a , b) ;
		a = 17
		b = 10

(b)	int x = 45 ;
	if (x + 20 * 7 = = 455)
	print f ("Look's Good");
	else
	print ("Hope for the Best");
		Look's Good

(c)	Char c1 = 'Y' , c2 = 'N' ;
	int n1 = 5, n2 = 9;
	n1 = n1 + 1;
	c1=c2;
	if (n1 = = n2 && c1 = = c2)
		6 < 9 and N=N

	print ("%d = %d and %c = %c" , n1, n2, c1, c1);
	else
	if (n1 < n2 && c1 = = c2)
	printf ("%d < %d and %c = %c", n1, n2, c1, c2);
	else
	printf ("Better Luck Next Time !");	

(d)	int a = 34, b = 32, c = 7, d = 15;
	a = b + c + d;
	if (a < 100)
	a = a * 2;
	b = b * c;
	c = c + d;
	if (a >b && c = = d)
	{
	c = d
	b = c;
	a = b;
	}
	else
	if (a > b && c > d || b>=d+c)
	{
	d = c * c;
	a = b * b;
	}
	printf ("a = %d, b=%d, c=%d, d=%d", a, b, c, d);
		a = 50176, b = 224, c = 22, d = 484

e)	int x = 5, y = 7 z = 9;
	if (x % 2 = = 0)
	x ++;
	else
	x = y + z;
	printf (" x = %d\n", x);
	if (x % 2 = = 1 && y % 2 = = 1 && z % 2 = = 1)
	printf ("All are Odd");
	if (x > y || x < z)
	{
	if (x > y)
	y ++;
	else
	if (x < z)
	print ("x = %d, y = %d, z = %d" , x, y, z);
		x = 16
		x = 16, y = 8, z = 9

PROGRAMMING EXERCISES (A.B)

EXERCISE 1

Write a program that takes two integers as input and tells whether first one is a factor of the second one?

```
# include <stdio.h>
        void main( )
        {
            int num_1, num_2;
            printf("enter first integer ");
            scanf("%d", &num_1);
            printf("enter second integer ");
            scanf("%d", &num_2);
            if(num_2%num_1==0)
            printf("Yes first one is the factor of second one");
            else
            printf("No first one is not the factor of second one");
        }
```

EXERCISE 2

Write a program that takes a number as input and displays "YES" if the input number is multiple of 3, and has 5 in unit's place e.g. 15, 75.

```
# include <stdio.h>
            void  main  ( )
        {
            int num;
            printf("input a number that has 5 in unit place (e.g. 15,75) ");
            scanf("%d", &num);
            if(num%3==0)
            printf("yes the input number is a multiple of 3");
            else
            printf("No the input number is not a multiple of 3");
        }
```

EXERCISE 3

Following is the list of discounts available in "Grocery Mart".

Total Bill	Discount
1000	10%
2500	20%
5000	35%
10000	50%

Write a program that takes total bill as input and tells how much discount the user has got and what the discounted price is.

```
# include <stdio.h>
                    void main( )
              {
                    float total_bill , discount;
                    printf("Enter your Total bill.");
                    scanf("%f", & total_bill);
                    if(total_bill>=1000 && total_bill<2500)
                    {
                       discount=total_bill*10/100;
                    printf("discounted price=%f", discount);

                    }
                    else if(total_bill>=2500 && total_bill<5000)
                    {
                       discount=total_bill*20/100;
                       printf("discounted price=%f", discount);
                    }
                    else if(total_bill>=5000 && total_bill<10000)
                    {
                       discount=total_bill*35/100;
                       printf("discounted price=%f", discount);
                    }
                    else if(total_bill>=10000)
                    {
                       discount=total_bill*50/100;
                 printf("discounted price=%f", discount);
                    }
              }
```

EXERCISE 4

Write a program that takes as input, the original price and sale price of a product and tells whether the product is sold on profit or loss. The program should also tell the profit/loss percentage.

```
#include <stdio.h>
void main()
{
                    float cost_price,selling_price, amount;
                       float profit_per, loss_per;
                    printf("Enter cost price: ");
                    scanf("%f", &cost_price);
                       printf("Enter selling price: ");
                    scanf("%f", &selling_price);
                    if(selling_price > cost_price)
                    {
                    amount = selling_price - cost_price;
                    profit_per=((amount*100)/cost_price);
                    printf("Profit = %f", amount);
                    printf("\nProfit_prcentage = %f", profit_per);
```

```
                    }
                    else if(cost_price > selling_price)
                    {
                    amount = cost_price - selling_price;
                    loss_per=((amount*100)/cost_price);
                    printf("Loss = %f", amount);
                    printf("Loss_prcentage = %f", loss_per);
                    }
                    else
                    {
                    printf("No Profit No Loss.");
                    }
}
```

EXERCISE 5

Write a program that takes as input, the lengths of 3 sides of a triangle and tells whether it is a right-angle triangle or not. For a right-angled triangle,
Hypotenuse^2= base^2 + height^2

```
# include <stdio.h>
            void main ( )
      {
            float side1,side2,side3;
            printf("Enter three sides of a triangle ");
            scanf("%f%f%f", &side1, &side2, &side3);
if(side1*side1 == side2*side2+side3*side3|| side2*side2 == side1*side1+side3*side3||
side3*side3==side1*side1+side2*side2)
                    printf("yes");

            else
                    printf("No");
      }
```

EXERCISE 6

Following is the eligibility criteria for admission in an IT University.

- **At least 60% marks in Matric.**
- **At least 65% marks in Intermediate (Pre-Engineering or ICS)**
- **At least 60% marks in entrance test**

Write a program that takes as input, the obtained and total marks of Matric, Intermediate and Entrance Test. The program should tell whether the students is eligible or not.

```
# include <stdio.h>
            void main ( )
            {
                    float obt_marks_matric, total_marks_matric;
                    float obt_marks_inter, total_marks_inter;
                    float obt_marks_entry_test, total_marks_entry_test;
                    float matric_prc, inter_prc, entry_test_prc;
                    printf("Enter Total marks of matric exams");
                    scanf("%f", &total_marks_matric);
                    printf("Enter obtained marks of matric exams");
                    scanf("%f", &obt_marks_matric);
                    printf("Enter Total marks of Intermediate exams");
                    scanf("%f", &total_marks_inter);
```

```
                printf("Enter obtained marks of Intermediate exams");
                scanf("%f", &  obt_marks_inter);
                printf("Enter Total marks of Entrance Test");
                scanf("%f", & total_marks_entry_test);
                printf("Enter obtained marks of Entrance Test exams");
                scanf("%f", & obt_marks_entry_test);
                matric_prc= obt_marks_matric/total_marks_matric*100;
                inter_prc= obt_marks_inter/total_marks_inter*100;
                entry_test_prc= obt_marks_entry_test/total_marks_entry_test*100;
                if(matric_prc>=60 && inter_prc>=65 && entry_test_prc>=65)
                printf("You are eligible for admission");
                else
                printf("You are not eligible for admission");
        }
```

EXERCISE 7

Write a program that calculates the bonus an employee can get on the following basis:

Salary	Experience with Company	Bonus Tasks	Bonus
10000	2 year	5	1500
10000	3 year	10	2500
25000	3 year	4	2000
75000	4 year	7	3500
100000	5 year	10	5000

The program should take as input, experience and number of bonus tasks of the employee. The program should display the bonus on the screen.

```
# include <stdio.h>
                void main ( )
                {
                        int salary, experience, bonus_tasks;
                        printf("Enter your current salary.");
                        scanf("%d", & salary);
                        printf("enter your experience with company in years.");
                        scanf("%d", & experience);
                        printf("Enter number of bonus tasks of the employee.");
                        scanf("%d", & bonus_tasks);
                        if(salary==10000 && experience==2 && bonus_tasks==5)
                        printf("Your calculated bonus = 1500");
                        else if(salary==10000 && experience==3 && bonus_tasks==10)
                        printf("Your calculated bonus =2500");
                        else if(salary==25000 && experience==3 && bonus_tasks==4)
                        printf("Your calculated bonus =2000");
                        else if(salary==75000 && experience==4 && bonus_tasks==7)
                        printf("Your calculated bonus =3500");
                        else if(salary==100000 && experience==5 && bonus_tasks==10)
                        printf("Your calculated bonus =5000");
                        else
                        printf("Invalid Input");
                }
```

ANSWER KEY

3.1 CONTROL STATEMENT

1	2	3	4	5
A	A	B	B	D

3.2 SELECTION STATEMENT

3.2.1 IF STRUCTURE

3.2.2 IF-ELSE STRUCTURE

3.2.3 NESTED SELECTION STRUCTURES

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