Short Questions

1. **What is conditional logic?**

Ans. All these decisions are taken on the basis of condition. If the condition is true, we perform the specified task, otherwise we do not. Sometimes, if the condition is not true then we perform some other task. This is called conditional logic.

2. **What is meant by Control Statements?**

Ans. The flow of program execution is controlled through control statement.
There are three types of control statements in C language.
1. Sequential Control Statements
2. Selection Control Statements
3. Repetition Control Statements

3. **What are Selection Statements?**

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

4. **What do you know about Sequential Control?**

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

5. **What is meant by Condition?**

Ans. A condition could be any valid expression including arithmetic expressions, relational expressions, logical expressions, or a combination of these.

6. **What are selection statements? How many types of selection statements?**

Ans. 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

7. **Why do we need selection statements?**

Ans. 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. Selection statements allow a program to test several conditions, and execute instructions based on which condition is true. That's why we need selection statements.

8. **What is associated code?**

Ans. The associated code in if statement is any valid C language set of statements.

9. **Differentiate between sequential and selection statements.**

Ans.

Sequential statements	Selection statements
Sequence of statements are written in order to accomplish a specific activity. So sequential statements are executed in the order they are specified in the program.	The statements which help us to decide which statements should be executed next, on the basis of conditions, are called selection statements.

10. **Which control is the default control structure in C language?**

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

11. **Differentiate between if statement and if else statement with an example.**

Ans.

if statement	if else statement

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

12. What is nested selection structures?

Ans. Selection statements within selection statements are called nested selection structures. General structur an if-else statement given below:

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

13. Write the structure of if statement.

Ans. if statement has the following structure in C language:

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

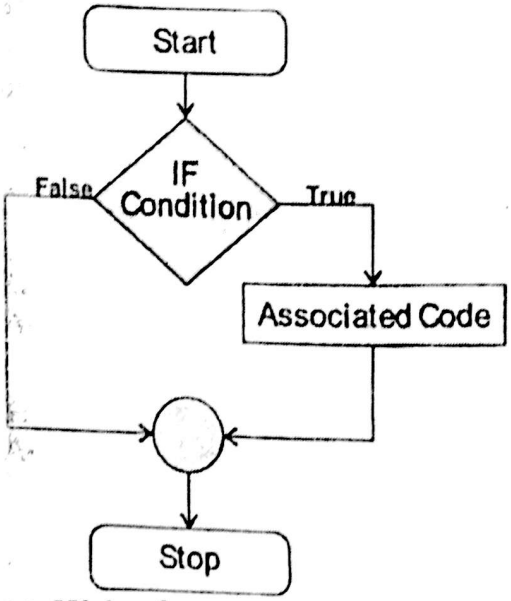
14. Write a program in C language that takes the percentage of student as an input and displays "PAS if the percentage is above 50.

Ans.

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

15. Draw flowchart of if statement.

Ans.



16. Write the structure of if-else statement.

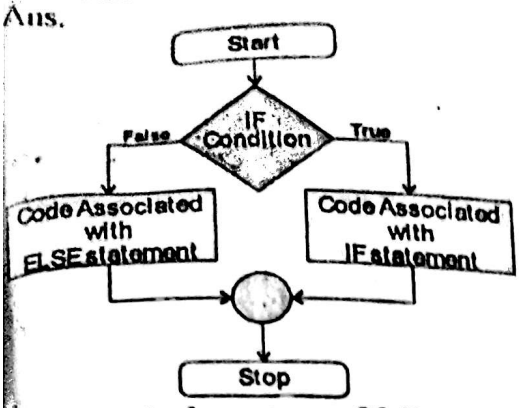
Ans. General Structure of the if-else statement is as follows:

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

CamScanner

17. Draw flowchart of if-else statement.

Ans.



18. What is the output of following code?

```
#include<studio.h>
void main ()
{
int a = 18;
if (a % 2 = = 0);
    {
        printf("Even value.");
     }
else
        printf("Odd value.");
}
```

Ans. Even value.

19. 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

Ans.

```
#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");
}
```

20. What are compound statements?

Find ... For Online lectures, Visit : www.youtube.com/ComputerTeacherOfficial

CamScanner

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

21. Name the types of control structure?

Ans. 1. Sequence Control statements

2. Selection Control statements

3. Repetition / Iteration / Loops Control statements

CamScanner

CH # 3 Conditional Logic

Q1. What are control statements? Also discuss its different types.

Ans. Control statements: These are the statements used to control the flow execution in a program or function. These instructions enable us to group individual instructions into a single logical unit with one entry point and one exit point. All programs use control statements to implement the program logic. There are three types of control structures:

1. Sequence Control statements
2. Selection Control statements
3. Repetition / Iteration / Loops Control statements

Sequence The instructions are executed in a sequence in which they are written in the program. It is default flow of the program. The execution of the program starts from first instruction and all instructions are executed one by one in a sequence. **Example:**

{
Statements
St – 1
St – 2
}

Selection A structure that selects which statement or block of statements is to execute. In selection structure instruction are divided into two or more groups and the selection is done after testing a condition.
There are two types of selection statements
1. if statements 2. if – else statement

Repetition/ iteration/ loops: These statements are used to repeat a set of statements as long as condition is true.
There are two types of loop statements
1. for-loop 2. while loop 3. do-while loop

Q2. What is if statement? Explain its structure, syntax and give example.

Ans. if is a keyword in C language. It is used to execute or ignore a set of statements after testing a condition. A condition is a relational or logical expression and it produces either true (means 1) or false (means 0) result. If the condition is true, then the block of statement is executed and if the condition is false then the block of statement is ignored and the control is transferred to the next statement after if statement.

Syntax: if (relational or logical condition)

```
{
Block of statements
}
next statement after if
```

Example: Even or Odd Number

```
#include <stdio.h>
void main ( )
{
int n;
printf("\n Enter any Number:");
scanf("%d", &n);
if(n%2==0)
{
        printf("\n The Number is Even ");
        printf("\n You are doing a great job");
}
}
```

Output

Enter any number 4

CamScanner

The Number is Even
You are doing a great job

Structure of if statement

If statement has the following structure in C language:

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

Here is a brief description of different components involved in the general structure of if statement.

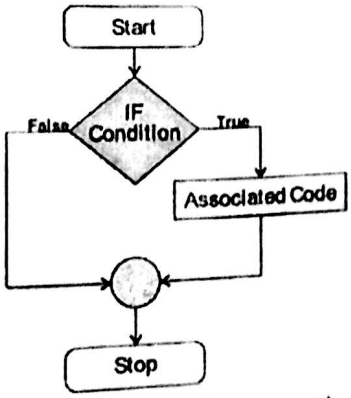
1. In the given structure, if is a keyword that is followed by a condition inside parentheses ().
2. A condition could be any valid expression including arithmetic expressions, relational expressions, logical expressions, or a combination of these. Here are a few examples of valid expressions that can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

a-	10	
b-	6 + 4	(true)
c-	1-1	(true)
d-	8>3	(false)
e-	6= = 10	(true)
f-	!(4>7)	(false)
g-	(5 >1) &&(11 < 9)	(true)
h-	(5 > 4) || (8 < 10)	(false)
		(true)

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. It is demonstrated through the following examples.

Because, when value 12 is divided by 2, it gives a remainder equal to 0, so the condition inside if parentheses is true. As both the printf statements are inside {} block, so both the statements get executed. Now look at the following example:

As the condition inside the if parentheses is false, and the statements following the if statement are not inside {} block, so only the 2nd statement is executed because without a {} block, only 1statement is considered to be associated with the if statement.

Q3. What is if-else statement? Explain its structure, syntax and give example.

Ans. It is similar to if statement i.e. It is also used to execute or ignore a set of statements after testing a condition. It is a two way decision making statement. In if-else statement one condition and two blocks of statements are given.

Find me on YOUTUBE For Online lectures, Visit : www.youtube.com/ComputerTeacherOfficial

CamScanner

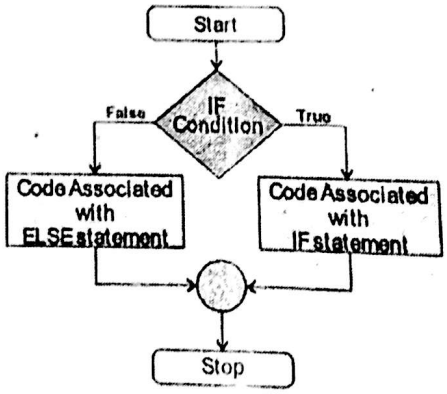
Syntax:

if (relational or logical condition)
{
first Block of statements
}
Else
{
2nd block of statements
}
Next statements after if-else

Structure

1. In the given structure, if is a keyword that is followed by a condition inside parentheses ().
2. else is a keyword which is associated with if. If the condition is true then the first block of if statement is executed and 2nd block of else is ignored. And after executing the first block, the control is transferred to next statement after if-else structure. If the condition is false then the first block of statement is ignored and the 2nd block of statement is executed. And after executing the 2nd block of statement the control is transferred to the next statement after if-else structure.
3. A condition is a relational or logical expression and it produces either true (means 1) or false (means 0) result. 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.
4. If there are more than one instructions under *if statement* or *else statement*, enclose them in the form of a block. Otherwise, the compiler considers only one instruction under it and further instructions are considered independent.

Flowchart



Note : *An if statement may not have an associated else statement, but an else statement must have an if statement to which it is associated.*

If there are multiple statements under **if** or **else**, then they must be enclosed inside the {} block, otherwise compiler issues an error.

Example:

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

CamScanner

```
else
        printf("The variable a contains an odd value.");
```

Output:

The variable a contains an odd value.

Q4. What is nested if statement or nested selection structure? Explain with the help of example and flow chart.

Ans. if statement within another if statement is called nested if statement. It is used for multi-way decision making. Nesting can be done up to any level. Increase in nesting also increase the complexity of program. The else statement is optional. It may be used with outer or inner if statement.

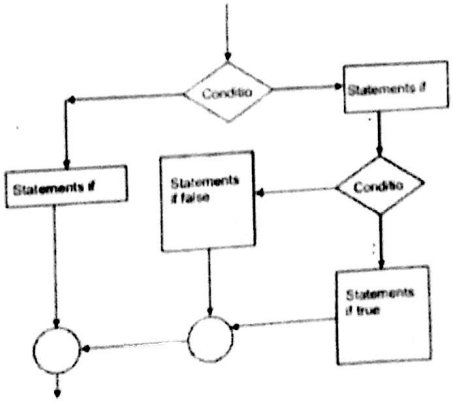
Syntax:

if (condition1 is true)
  if (condition2 is true)
  Associated code
  else
  Associated code
	if (conditiona1 is true)
	  if (condition2 is true)
	  Associated code
	else
	  if (conditions is true)
	  Associated code

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

Inside the inner if statements or if-else statements we can have even more if statements or if-else statements and so on. Conditional statements within conditional statements are also called nested selection structures.

Flowchart:



Example:

Program to find largest number using nested selection structure

CamScanner

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

CamScanner