

INU

Pg # 1

Name → Farhan Ali

ID → 12761

Course → Programming fundamental

Program → BS (Tele)

Exam → Final

Submitted to

Dr. Fazel-e-Malik.

(Q 1) what is the purpose of if statement  
 (a) Discuss its two different forms with example.

(Ans) The if statement is used to check a condition and if the condition is true, we run a block of statements (called the if block) else we process another block of statements (called the else block). The else clause is optional.

### Form of if statement

(A) If (condition)

```
int x, y;
```

```
scanf ("%d", &x);
```

```
y = (x > 5 ? 3 : 4);
```

The equivalent if statement will be

```
if (x > 5)
```

```
    y = 3;
```

```
    y = 4;
```

Cont—

(b) (if condition)

{

char a;

int y;

char &gt;&gt; a

y = (a &gt;= 'G' &amp;&amp; a &lt;= 'g' ? 1 : 0);

⇒ The three forms of if statement:

IF THEN ; IF THEN - ELSE and IF - THEN - ELSE IF

The simplest form of if statement associates a Boolean expression with a sequence of statement enclosed by keyword . THEN and END IF . The sequence of statement is executed only if the expression between true.

\* Expression is true

int test = 5;

if (test &lt; 10)

{

// codes

// codes after if

Expression is false

int test = 5;

if (test &gt; 10)

{

// codes

{

→ // codes after if

(Q1) write a C++ program to read  
(a) two numbers from keyboard and  
then find the largest number of them

```
#include <iostream>
using namespace std;
```

```
int main()
```

```
{
```

```
    int num1, num2;
```

```
    cout << "Enter first number:";
```

```
    cin >> num1;
```

```
    cout << "Enter second number:";
```

```
    cin >> num2;
```

```
    if (num1 > num2)
```

```
    {
```

```
        cout << "First number" << num1 << " is the largest";
```

```
    }
```

```
    else
```

```
        cout << "second number" << num2 << " is the  
largest";
```

```
    }
```

```
    return 0;
```

```
}
```

(Q2) what are logical operators  
 (a) Explain them.

(Ans) Logical Operators

A logical operator is a symbol or word used to connect two or more expressions such that the value of the compound produced depends only on that of the original expressions and on the meaning of the operator. Common logical operators include AND, OR and NOT.

\* Explanation

with most languages expressions that yield Boolean data type values are divided into two groups. one group the relational operators within their expressions and the other group uses logical operators within their expressions.

#6

The logical operators are often used to help create a test expression that controls program flow. This type of expression is also known as a Boolean expression.

⇒ Operator symbols and/or name vary with different programming languages

Language	AND	OR	NOT
C++	&&		!
C#	&&		!
Java	&&		!
JavaScript	&&		!
Python	and	or	not
Swift	&&		!

&& (logical AND)

- combine two conditions

- true if both conditions are true

if (gender == "M" && age >= 65)

cout << "Retirement age";

Logical OR

- true if either of conditions is true

if (semester Avg  $\geq$  90 || final Exam  $\geq$  90)  
 cout << "student grade is A";

Logical NOT, (logical negation)

- Return true when its condition is false and vice versa

if ( ! ( grade == 20 ) )  
 cout << "hello world";

Alternative

if ( grade != 20 )  
 cout << "hello world";

! False  $\rightarrow$  if (True)

! True  $\rightarrow$  if (False),

Q2 write a C++ program to get (b) temperature in Fahrenheit F and then find the Atmosphere according to the below rules:

• If temperature -----

-----  
-----

Fahrenheit then display - 'cool'

Sol

```

1  /*
2  * C++ program perform Temperature in Fahrenheit
3  */
4  #include <iostream>
5  using namespace std;
6
7  int main ()
8  {
9      float Fahrenheit, Temperature;
10
11     cout << "Enter the Temperature in Fahrenheit: ";
12     cin >> Fahrenheit;
13     Fahrenheit = (Celsius * 9.0) / 5.0 + 32;
14     cout << "The temperature in Fahrenheit: " << Celsius << endl;

```



```

15 cout << "The temperature in Fahrenheit: " << fahrenheit << endl;
16     return 0;
17 }

```

```

#include <stdio.h>
void main ()
{
    int temp;
    printf ("input days temperature : ");
    scanf ("%d", &temp);

    printf ("Freezing weather.\n");
    else if (temp < 10)
        printf ("cold weather.\n");
    else if (temp >= 30)
        printf ("Normal in temp.\n");
    else if (temp >= 40)
        printf ("Its Hot,\n");
    else
        printf ("Its very hot,\n");
}

```

(Q3) What does looping means?

(a) Explain different loops in C++.

(Ans) A loop is used for executing a block of statement repeatedly until a particular condition is satisfied in C++ we have three types of basic loops for while and do while.

### \* Loop in C++

⇒ Loop structure is used to execute a statement or statements repeatedly

Three major loops structure are

- While loop
- for loop
- do while.

Loop in General

→ Loop has a termination point → finite loop  
 - loop execution stops when the loop condition becomes false

→ Loop has a counter that counts number of iterations of that loop.

\* General form while loop

initialise loop counter;  
 while (test loop counter using a condition)

{  
 1 do this;  
 2 and this;  
 3 increment loop counter;

\* Points to remember

Loop body will keep on executing until the loop condition became false

→ When loop condition become false, the first statement after the while block will be executed.

⇒ Condition can be a single or compound.

while ( $i < 10$ )

while ( $i > 10$  &&  $j < 15$ )

while ( $i > 10$  && ( $b < 15$  ||  $c < 20$ )).

⇒ Statement within loop body can be single line or block of statement.

⇒ In case of single line parentheses are optional.

while ( $i < 10$ )

$i = i + 1;$

is same as while ( $i < 10$ )

{  
 $i = i + 1;$

}

(Q3) write a C++ program to read a  
 (b) number from keyboard and then  
 determine whether it is even or odd number

1. /\*
2. \* C++ program to check if given integer is even or odd
3. \*/
4. #include <iostream>
5. using namespace std;
- 6.
7. int main ()
8. {
9. int number, remainder;
- 10.
11. cout << "Enter the number : ";
12. cin >> number;
13. remainder = number % 2;
14. if (remainder == 0)
15. cout << number << " is an even integer" << endl;
16. else
17. cout << number << " is an odd integer" << endl;
- 18.
19. return 0;
20. }

```
$ g++ main.cpp
```

```
$ ./a.out
```

```
Enter The number : 3
```

```
3 is is an odd integer
```

```
$ ./a.out
```

```
Enter The number : 10
```

```
10 is an even integer.
```

#15

(Q4) (a) what is the purpose of using break and continue statements?

(Ans) Break Statement

- There are situations where we want to jump out of a loop instantly, without waiting to get back to the conditional test
  - The keyword break allows us to do this
  - When break is encountered inside any loop, control automatically passes to the first statement after the loop.
  - A break is usually associated with an if
  - The key word break. breaks the control only from the loop in which it is placed.
- \*Break Statement in nested loop

```
main()
{
    int i = 1, j = 1;
    while (i++ <= 100)
    {
```

Continue →

```

while (j++ < 20)
{
    if (j == 150)
        break; ←
    else
        cout << "\n", i, j << "\n";
}

```

```

}
}

```

### \* The continue statement.

Continue statement allows to take to the beginning of the loop, bypassing the statement inside the loop, which have not yet been executed.

→ A continue is usually associated with an if

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

```
main()
```

```
{ int i, j;
```

```
for (i = 1; i <= 2; i++)
{
```

```
for (i = 1; j <= 2; j++)
```

```

{
    if (i == j)
        continue;
    cout << "\n\n", i, j << "\n";
}
}
system ("pause");
}

```



(QW) Write a C++ program to find  
 (b) the sum of the following numbers.  
 $1 + 2 + 3 + \dots + 10$

1. #include <iostream
2. using namespace std;
3. int main ()
4. {
5. int i, sum = 0;
6. cout << "\n\n Find the first 10 natural numbers: \n";
7. cout << " \_\_\_\_\_ " \_\_\_\_\_ " \_\_\_\_\_ \n";
8. cout << "The natural numbers are: \n";
9. for (i = 1; i <= 10; i++)
10. {
11. cout << i << " ";
12. sum = sum + i;
13. }
14. cout << "\n The sum of first 10 natural numbers is
15. {  
 " << sum << endl;

Sample output

Find the first 10 natural numbers;

The natural number are.

1 2 3 4 5 6 7 8 9 10

The sum of first 10 natural number; 55.

(Q5) Explain the following with proper example.

(a) C++ Character Set.

In C++ character set is a set of all valid characters that can be used in a C++ program. Character set is used to specify the characters or symbols recognised by the language. Character set is a set of all valid characters that can be used to form words, numbers and expressions in source programs.

\* Example

```

/* C++ Program = Add Two Number */
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();
    int a, b, sum;
    cout << "Enter two number : ";
    cin >> a >> b;
    sum = a + b;
    cout << "sum of the two number is " << sum;
    getch();
}

```

## (b) Constant :-

# 20

Constant refer to fixed value that the program may not ~~after~~ alter and they are called literals. Constant can be of any of the basic data types and can be divided into integer numerals, floating point numbers, characters, strings and Boolean values.

## (c) variables :-

A symbol for a value we don't know yet. It's usually a letter like  $x$  or  $y$ .

### Example.

$$\text{In } x + 2 = 6$$

$x$  is the variable.

(d) keyword :-

#21

The definition of a keyword is a signified word or a word used to bind information when researching;

When you are looking for resume example examples and you type "resume examples" into Google. The word "resume examples" are an example of keyword.

(e) Relational operators :-

A relational operator is a programming ~~operator~~ language construct or operator that test or defines some kind of relation between two entities. These include numerical equality

e.g.  $5 = 5$

and inequalities

e.g.  $4 \geq 3$

Relational operators can be seen as special case of logical predicates