**Notes of Important programming Fundamental Questions**

**Functions**

**What is a function?**

A function is a named block of code that performs some action. The statement written in a function are executed when is called by the name. Each function has a unique name. Functions are the building blocks of C++ programs. Functions are easier to write, modify and maintain. Functions can be reused and reduces programming time.

**What is function header?**

The first line of function definition is known as function header. The function header consists of return type, function name and parameter.

**What is function body?**

The set of statements which are executed inside the function is known as function body. The body of function appears after function header. The statements are written in curly braces {}. The variable declaration and program logic are implemented in function body.

**What is function declaration or function prototype?**

Function declaration is a model of a function. It is also known function prototype. It provides information to compiler about the structure of the function to be used in program. It consists of function name, function return type and number and type of parameters. It is terminated with semicolon.

**What is a function called?**

The statement that activates a function is known as function call. A function is called its name. Function name is followed by necessary parameters in parentheses. If there are many parameters, these are separated by commas.

**What is local variable?**

A variable declared inside a function is known as local variable. Local variable can be used only in the function in which it is declared. The lifetime of local variable starts when control enters the function in which it is declared. Local variable is automatically destroyed when control exist function.

**What is global variable?**

A variable declared outside any function is known as global variable. Global variable can be used by all functions in the program. Global variable exist in the memory as long as the program is running. These variables are destroyed from the memory when the program terminates.

**What is scope of local variable?**

The area where a variable can be accessed is known as scope of variable. Local variable can be used only in the function in which it is declared. If a statement accesses a local variable that is not in scope, the compiler generates a syntax error.

**What is scope of global variable?**

Global variable can be used by all functions in the program. It means that these variables are globally accessed from any part of the program. Normally, global variables are declared before main function.

**What is lifetime of local variable?**

The time period for which a variable exist in the memory is known as the lifetime of variable. Different types of variable have different lifetime.

The lifetime of local variable starts when the control enters the function in which it is declared. Local variable is automatically destroyed when the control exists from the function and its lifetime ends. When the lifetime of a local variable ends, the value stored in this variable also becomes inaccessible.

**What is lifetime of global variable?**

Global variables exist in the memory as long as the program is running. These variables are destroyed from the memory when the program terminates. These variables occupy memory longer than local variables. So, global variables should be used only when these are very necessary.

**Loops**

**What is loop? What are two uses of loop?**

A control structure that executes a statement or number of statement repeatedly is known as loop. Loops can be used to execute a statement or number of statement for a specified number of time. Loops can also be used to access a sequence of value such as 1,2,3 so on.

**What is difference between while and do-while loops?**

In while loop, condition comes before the body of the loop. In do-while loop, condition comes after the body of the loop. If condition is false in the beginning, while loop is never executed. Do-while is executed at least once even if condition is false in the beginning.

**Difference between counter and conditional loops?**

Counter loop depends on the value of a variable known as counter variable. The value of counter variable is increment or decrement each time the body of the loop is executed. This loop terminates when the value of counter variable reaches a particular value.

Conditional loop depends on special value known as sentinel value. Sentinel value indicates that the loop should continue or terminate. For example, a loop may execute while the value of a variable is not -1. Here -1 is the sentinel value used to terminate loop.

**What is nested loop?**

A loop within a loop is called nested loop. In nested loop, the inner loop is executed completely with each change in the value of counter variable of outer loop. Any loop can be used as inner loop of another loop.

**What is infinite loop?**

A loop in which the ending condition never occurs is called indefinite loop. It repeats forever unit the user intervenes to stop the loop.

Describe the syntax of while loop with example.

The syntax of while loop is as follows:

While (condition)

 Statement(s);

**Example:**

int a=0;

while(a<10)

{

cout<<”Adeel Rasheed”;

a++;

}

**Arrays**

**Define an array?**

An array is a group of consecutive memory locations with same name and type. All these

Memory locations have one collective name and type. The memory locations in the array are known as elements of array. The total number of elements in the array is called its length.

**How an element is is accessed in array?**

Each element in the array is accessed with reference to its position of location in the array. The position is called index or subscript. The index of first element is 0 and index of last element is length -1. The value of the index is written in brackets along with the name of array.

**How subscript variables are written?**

A subscript variable is written in brackets [] with array name.

**Distinguish between 1-D array and 2-D array?**

1-D array consists of only single row or single column. Each element of this array can be accessed using one index value. 2-D array consists of multiple row and columns. Each element of this array is accessed using two index value.

**Explain array manipulation?**

The process of performing different operations on array is called array manipulation. Different operations can be performed using array such as finding an element, comparing two arrays, finding the minimum or maximum value in array and sorting an array etc.

**Conditional Structure**

**What is control structure?**

A statement used to control the flow of execution in a program or function is called control structure. The control structures in C are used to combine individual instruction into a single logical unit. The logical unit has one entry point and one exit point.

**What are the basic control structures for written programs?**

The basic control structures for written programs are sequence, selection and repetition.

**What is Relational Operations?**

The relational operators are used to specify conditions in programs. A relational operator compares two values. It produces result as true or false. The relational operators are sometime called the conditional operators or comparison operators as they test conditions that are either true or false. C++ provides different relational operators. These are >, <, ==, >=, <=, and !=.

**What is relational expression?**

Relational expression is a statement that uses relational operators to compare two values. Examples of relational expressions are A>B, A<B, A>=B, A<=B, A==B, and A!=B.

**What is Compound Condition?**

A type of comparison in which more than one conditions are evaluated is called compound condition. It is used to execute a statement or set of statements by testing many conditions. For example, a program inputs two numbers. It displays OK if one numbers is greater than 100 and second number is less than 100. Compound condition is executed by using logical operators.

 **What happens if break is missed in case block?**

If break is not used, all case blocks coming after matching case will also be executed.

**Why break statement is used in a switch () structure?**

The break statement in each case label is used to exit from switch body. If break is not used, all case blocks coming after matching case will also be executed.

**Why should you use a default label in a switch statement?**

The default block is used to execute a statement or set of statements when the result of expression in switch statement does not match with any case. If makes sure that switch executes properly if the selector is not in the range of case labels.

**What is conditional operator?**

Conditional operator is a decision-making structure. If can be used in place of simple “if-else” structure. It is also called ternary operator because it used three operands. The syntax of conditional operator is as follow: (condition)? True-case statement : false-case statement: