February 1, 2016

INTRODUCTION

Q.What does computer-programming mean? Exam:Acce-2014
Q.What is mean by computer program and programming language? Exam:Acce-2013
Q.What is Programming Language?
Ans:
Computer program:

To process of particular set of data the computer must be given an appropriate set of instructions called a program . these instruction are entered into the computer and then stored in a portion of computers memory.

Computer programming is a process that leads from an original formulation of a computing problem to executable **computer** programs.

Programming language: A **programming language** is a special **language programmers** use to develop software programs, scripts, or other sets of instructions for computers to execute. The following is an index of the different **programming** and scripting **languages** currently listed at mine.

Q.Discuss different kind of Programming language? Exam:Acce-2013

Language types
1.From

- Machine and assembly languages
- Algorithmic languages
  - FORTRAN
  - ALGOL
  - LISP
  - C
- Business-oriented languages
  - COBOL

2. Above the level

- **High level** - Ada , Modula-2 , Pascal, COBOL, FORTRAN, BASIC, Java, C++.
- **Middle level** - C, FORTH, Macro-assemble.
- **Low level** - Assembler.

Q.why c is called middle level language? Exam:Acce-2013

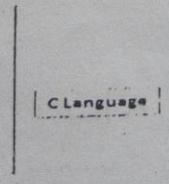
Middle Level Language:

C Programming bridges gap between Machine Understandable Machine Level language and more conventional High level languages.

Why C is Middle Level Language:

1. C programming language support the low level language i.e. Assembly Language..
2. C language also gives the facility to access memory through pointer.
3. Its combines the elements of high-level languages with the functionalism of assembly language.
4. Using inline assembly language feature in C we can directly access system registers.
5. C Programming also Supports high Level Language Features.

So, C language neither a High Level nor a Low level language but a **Middle Level Language**.



Q.Why compiler Need?
Ans:

The term compiler refers to the way in which a program is executed. in theory any programming language can be compiled or interpreted.For example BASIC is usually interpreted and c is usually compiled. The way a program is executed is not defined by the language in which it is written compilers are simply sophisticated programs that operate on our program source code.

Q.Why to use C?
Ans:

C was initially used for system development work, in particular the programs that make up the operating system. C was adopted as a system development language because it produces code that runs nearly as fast as code written in assembly language. Some examples of the use of C might be:

- Operating Systems
- Language Compilers
- Assemblers
- Text Editors
- Print Spoolers
- Network Drivers
- Modern Programs
- Databases
- Language Interpreters
- Utilities

Q.What is Compiler? Exam:ACCE-2012,ICE-CSE-
Ans:

A compiler is a computer program that transforms source code written in a programming language (the source language) into another computer language (the target language)
or
with the latter often having a binary form known as object code.

Q.What is Algorithm? Exam:ACCE-2012,ICE-CSE- MSE,APPE-,CSE-2011
Ans:

Algorithm is a finite set of steps to solve problem by computer.
Or An algorithm (pronounced AL-go-rith-um) is a procedure or formula for solving a problem.
Or An algorithm is a sequence of unambiguous instructions for solving a problem, i.e., for obtaining a required output for any legitimate input in a finite amount of time

Example: If we want largest number from 2 number .that time our algorithm is

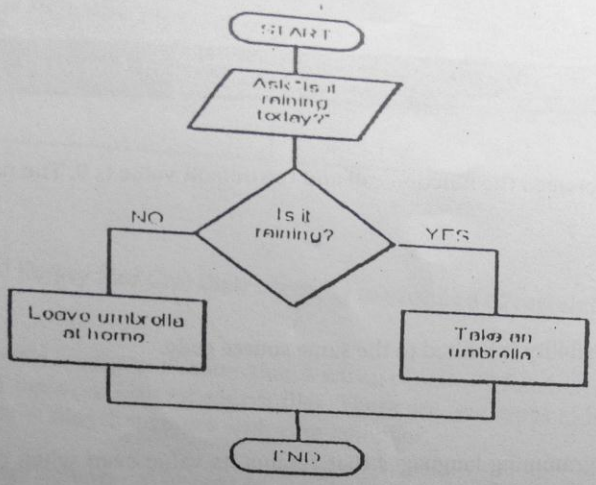
- Input 2 number
- Choose largest number
- Then print largest number

Q.What is flowchart?
Ans:

A flowchart is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows.
or
Flowchart is a graphical representation use to solve a particular problem.

Symbol	Name	Function
	Start/end	An oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Input/Output	A parallelogram represents input or output
	Process	A rectangle represents a process
	Decision	A diamond indicates a decision

bole of flowchart:
mple:



Q. Discuss the similarities and dissimilarities of algorithm and flowchart. Exam-2011

Ans:

Similarities:

- Flow chart is very important tool for developing algorithm and program . It is pictorial representation of step by step solution of a problem.
- Programmer often uses it as a program planning tool for visually organising step necessary to solve a problem. It uses boxes of different shapes that denotes different type of instruction.
- While making a flow chart a programmer need not to pay attention on the elements of the programming language ,he has to pay attention to the logic of solution to the problem

- The term algorithm refers to the logic.it is step by step description how to arrive at the solution to the problem.algorithm is define as sequence of instruction that when executed in the specified sequence the desired results are obtained .
- The set of rules that define how a particular problem can be solved in finite number of steps is known as algorithm.
- A good algorithm help us to create a good program.

Dissimilarity:

- An algorithm is a description of how to carry out a process. An algorithm lists the steps that must be followed to complete the process. Algorithms can be described in English but such descriptions are often ambiguous and open to misunderstanding. Therefore various formal methods of describing algorithms have been developed. The simplest of these is the flowchart.
- A flowchart consists of a sequence of instructions linked together by arrows to show the order in which the instructions must be carried out. Each instruction is put into a box. The boxes are different shapes depending upon what the instruction is.

Q.State whether each of the following sentences are true or false. If a sentence if false then explain the reason?

- When an array is passed as an argument to a function, a pointer is passed Ans:YES

Q.What is a static variable? APPE-CSE-

Ans:

A static local variables retains its value between the function call and the default value is 0. The following function will print 1 2 3 if called thrice.

```
void f() {
  static int i;
  ++i;
  printf("%d ",i);
}
```

If a global variable is static then its visibility is limited to the same source code.

Q.Define static variable. CSE-2011

Ans:

A kind of local variable in certain programming languages that retains its value even when program execution moves outside its scope.

Static variables have a fixed location in the data section of the program's address space whereas automatic variables are typically allocated on the stack.

Q.What is a preprocessor? CSE-MSE,APPE

Ans:

Preprocessor is a directive to the compiler to perform certain things before the actual compilation process begins.

Q.Define preprocessor?

Ans:

The C preprocessor or cpp is the macro preprocessor for the C and C++ computer programming languages. The preprocessor provides the ability for the inclusion of header files, macro expansions, conditional compilation, and line control.

In many C implementations, it is a separate program invoked by the compiler as the first part of translation.

Example:

```
#include <stdio.h>
int main(void)
{
  printf("Hello, world!\n");
  return 0;
}
```

the preprocessor replaces the line #include <stdio.h> with the text of the file 'stdio.h', which declares the printf() function among other things.

Q. What are the differences between compiler and Interpreter?

OR compare between compiler and Interpreter? Exam: CSE-Acce-2014,13,12

Ans:

Difference between Compiler and Interpreter

Comparison of compiler and Interpreter are as follows:

Compiler	Interpreter
Scans the entire program and translates it as a whole into machine code	Translates program one statement at a time.
Intermediate Object Code is Generated	No Intermediate Object Code is Generated
Conditional Control Statements are Executes faster	Conditional Control Statements are Executes slower
Memory Requirement : More (Since Object Code is Generated)	Memory Requirement is Less
Program need not be compiled every time	Every time higher level program is converted into lower level program

6Errors are displayed after entire program is checked	Every Errors are displayed for every instruction interpreted (if any)
7Example : C ,C++Compiler	Example : BASIC, Python, Ruby

Q.What is header file and library file? Give their necessity to execute a c Program?
Ans:

Header file: A header file is a file with extension .h which contains C function declarations and macro definitions to be shared between several source files. There are two types of header files: the files that the programmer writes and the files that comes with your compiler.

Library file: In computer science, a library is a collection of non-volatile resources used by computer programs, often to develop software. These may include configuration data, documentation, help data, message templates, pre-written code and subroutines, classes, values or type specifications.

Necessity header and library file to execute a c Program:
Necessity of header file:

You need "header files" when you are using an external library and when you want to avoid recompiling this library. It means that that you have to declare the objects of the library to your code somehow Various solution have been used. In C: you put the definition (only) in "header files" [double cos(double)].

Finally you need header files or an equivalent technology, when you need to access objects (variables, constants, functions, labels,...) in an external library that you do not want to recompile, modify or interfere with.

Necessity of library file:

We include header file with .h extension in our program and its just the definition but the actual implementation is defined in library files and this is done at linking stage this is what people say . A library file is the actual *executable code* that does the work as specified in that header file. This is linked in by the *linker* to provide the actual functionality (the _definitions rather than just the declarations

Q. Describe Different type of Header, Keyword and its tasks?Exam:ICE-2013,CSE,APPE-
Ans:

There are some header file in C language.Example:

- #include <stdio.h>
  - Stdio abbreviation of Standard input output.
  - Preprocessor directive
  - Tells computer to load contents of a certain file

#include<stdio.h>
#include<conio.h>

- "printf( )" is used -
  - (a) 1.To show the text display
  - (b) 2.To show the display the content of the variable
- "clrscr( )" is used to clear the screen.
- "Getch( ) is used to get the output & to comeback to the program
- "scanf( )" is used to get the input from the user.
  - it is stored in header file like dictionary i.e. <stdio.h>
- int main()
  - C++ programs contain one or more functions, exactly one of which must be main

- Parenthesis used to indicate a function
- Int means that main "returns" an integer value
- Braces ({ and }) indicate a block
- The bodies of all functions must be contained in braces

❖ printf("Welcome to C!\n");
- Instructs computer to perform an action
- Specifically, prints the string of characters within quotes (" ")
- Entire line called a statement
- All statements must end with a semicolon (;)
- \n is the newline character

❖ scanf("%d", &integer1);
- Obtains a value from the user
- scanf uses standard input (usually keyboard)
- This scanf statement has two arguments
- %d - indicates data should be a decimal integer
- &integer1 - location in memory to store variable

❖ return 0;
- A way to exit a function
- return 0, in this case, means that the program terminated normally

Q.How a negative integer is stored.?
Ans:
Get the two's compliment of the same positive integer. Eg: 1101 (-5)
Step-1 – One's compliment of 5 : 1010
Step-2 – Add 1 to above, giving 1011, which is -5

Q.What is the purpose of extern storage specifier? CSE-2011
Ans:
Used to resolve the scope of global symbol.

```
Ex:
main() {
  extern int I;
  Printf("%d",i);
}
int i = 20;
```

Q.Explain the purpose of the function sprintf().
Ans:
Prints the formatted output onto the character array.

Q.What is the purpose of the keyword typedef? Exam:ICE-2013
Ans:
It is used to alias the existing type. Also used to simplify the complex declaration of the type.

Q.Can a program be compiled without main() function?
Ans:
Yes, it can be but cannot be executed, as the execution requires main() function definition.

Q.What is the difference between variable declaration and variable definition?
Ans:
Declaration associates type to the variable whereas definition gives the value to the variable.

Q.Does a built-in header file contains built-in function definition?
Ans:
No, the header file only declares function. The definition is in library which is linked by the linker.

Q.Explain modular programming.
Ans:

dividing the program in to sub programs (modules/function) to achieve the given task is modular approach. More generic functions definition gives the ability to re-use the functions, such as built-in library functions.

Q.How can you print a \ (backslash) using any of the printf() family of functions.
Ans:

Escape it using \ (backslash).

Q.What is typecasting?
Ans:

Typecasting is a way to convert a variable/constant from one type to another type.

Q.What is the default value of local and global variables?
Ans:

Local variables get garbage value and global variables get a value 0 by default.

Q.Explain the use of comma operator (,).
Ans:

Comma operator can be used to separate two or more expressions.
Eg: printf("hi") , printf("Hello");

Q.How many operators are there under the category of ternary operators?
Ans:

There is only one operator and is conditional operator (? :)

What is reminder for 5.0 % 2?
Ans:

Error, It is invalid that either of the operands for the modulus operator (%) is a real number.

Q.Where the address of operator (&) cannot be used?
Ans:

It cannot be used on constants.
It cannot be used on variable which are declared using register storage class.

Q.What do you mean by structured programming ? write down the general structure of a c program?
Q.Discuss structure of C Programming? Exam: ACCE-2013
Ans:

Structured Programming:

Structured programming is a **programming** paradigm aimed at improving the clarity, quality, and development time of a computer program by making extensive use of subroutines, block structures, for and while loops—in contrast to using simple tests and jumps such as the goto statement which could lead to "spaghetti code".

Or

Structured programming is a procedural programming subset that reduces the need for goto statements. In many ways, OOP is considered a type of structured programming that deploys structured programming techniques. Certain languages – like Pascal, Algorithmic Language (ALGOL) and Ada – are designed to enforce structured programming.

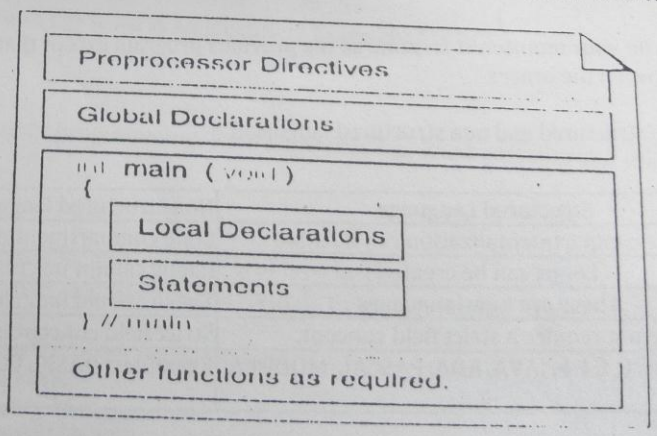


Fig: General Structure of C program.

Q.Briefly discuss about a simple c Program?

Ans:

A Simple C Program

Every C program must have one special function main (). This is the point where execution begins when the program is running. We will see later that this does not have to be the first statement in the program, but it must exist as the entry point. The group of statements defining the main () enclosed in a pair of braces ({}) are executed sequentially. Each expression statement must end with a semicolon. The closing brace of the main function signals the end of the program. The main function can be located anywhere in the program but the general practice is to place it as the first function.

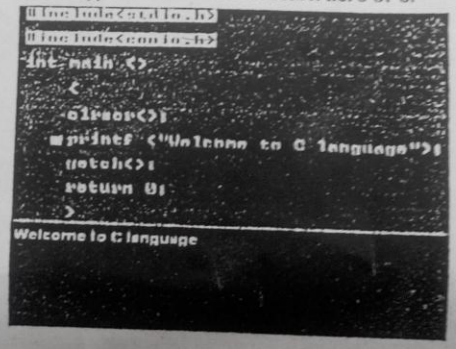
Here is an elementary C program.

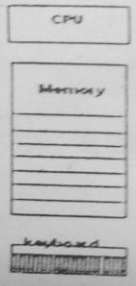
```
main ()
  {
  }
```

There is no way to simplify this program. main ( ) function should return zero or 0.

main( ) should be declared as

```
int main( )
 {
 ....
 ....
 return 0;
}
```





.For a much more interesting program, load the program

```
int main ()
  {
   printf
("Welcome
to C
language");
    return 0;
  }
```

and display it on your monitor. It is same as the previous program except that it has one executable statement between the braces.

Q.Different the structured and non structured language?

Ans:

Structured Language	Non structured language
Code compartmentalization can be done	Code compartmentalization cannot be done
Loops can be created	Loops cannot be created
These are new languages	These are old languages
Do not require a strict field concept.	Strict field concept is used mostly.
Examples: C, C++, JAVA, ADA, PASCAL, MODULA -2	Examples: BASIC, COBOL, FORTRAN

Q.what are the difference between structured programming and object oriented programming?

Acce-2014.CSE,APPE,MSE

Ans:

Structured Programming	Object Oriented Programming
Structured Programming is designed which focuses on process/ logical structure and then data	Object Oriented Programming is designed which focuses on data.

equired for that process.	
Structured programming follows top-down approach.	Object oriented programming follows bottom-up approach.
Structured Programming is also known as Modular Programming and a subset of procedural programming language.	Object Oriented Programming supports inheritance, encapsulation, abstraction, polymorphism, etc.
In Structured Programming, Programs are divided into small self contained functions.	In Object Oriented Programming, Programs are divided into small entities called objects.
Structured Programming is less secure as there is no way of data hiding.	Object Oriented Programming is more secure as having data hiding feature.
Structured Programming can solve moderately complex programs.	Object Oriented Programming can solve any complex programs.
Structured Programming provides less reusability, more function dependency.	Object Oriented Programming provides more reusability, less function dependency.
Less abstraction and less flexibility.	More abstraction and more flexibility.

Q.What is a token?
Ans:

A C program consists of various tokens and a token is either a keyword, an identifier, a constant, a string literal, or a symbol.

Q.Discuss about Token.
Ans:

In a passage of text individual words and punctuation marks are called tokens.

- In a C program the smallest individual units known as C tokens.
- C has 6 types of tokens namely:

1) Keywords
2) identifiers
3)constants
4)Strings
5)special symbols
6)operator

Q. Define key words and Identifier ? can you use a key word as an Identifier? Exam-2011,ICE,CSE
Ans;
Key Word :

Keywords are the reserved words used in programming. Each keywords has fixed meaning and that cannot be changed by user.
For example:

int *money*;

Here, int is a keyword that indicates, '*money*' is of type integer.
As, C programming is case sensitive, all keywords must be written in lowercase

- Keywords serve as basic building blocks for program statements.
- All keywords must be written in lower case.
- The underscore character is also permitted in identifiers.
- It is usually used a link between two words in long identifiers.

Points to remember

1. Keywords can be used only for their intended purpose.
2. Keywords can't be used as programmer defined identifier.
3. The keywords can't be used as names for variables.

Identifier:

Identifiers allow us to name data and other objects in the program. Each identified object in the computer is stored at a unique address.

or

Identifier is a name used to identify a variable, function, or any other user-defined item. An identifier starts with a letter A to Z or a to z or an underscore _ followed by zero or more letters, underscores, and digits (0 to 9).

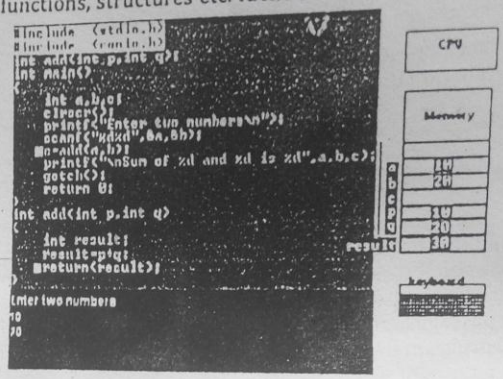
Or

Identifiers are names given to C entities, such as variables, functions, structures etc. Identifier are created to give unique name to C entities to identify it during the execution of program. For example:

int *money*;

int *mango_tree*;

Here, *money* is a identifier which denotes a variable of type integer. Similarly, *mango_tree* is another identifier, which denotes another variable of type integer.



We can not use a key word as an identifier

Q.Write the rules of naming variable? Exam –ACCE-2014,ICE-2013,CSE-2011

RULES FOR IDENTIFIERS/variables:

Ans:

(a) First character must be an alphabet.
(b) Must consist of only letters, digits or underscore.
(c) Only first 31 characters are significant.
(d) Cannot use keyword. Must not contain white space.
(e) C does not allow punctuation characters such as @, $, and % within identifiers.
(f) C is a case sensitive programming language.

Here are some examples of acceptable Identifiers:

mohd, zara, abc, move_name, a_123.
myname50 _temp, j, a23b9, retVal.
No we can not use keyword as identifier.

Q. What are the keywords in C ? what restriction apply to their use.

Ans:

Every C word is classified as either a keyword. The list of all keywords of C are listed bellow:

auto double int struct break else long switch case enum register typedef char extern return union const float short unsigned continue for signed void default goto sizeof volatile do if static while

All keywords have fixed meaning and these meaning can not be changed . the keyword are all lowercase.

Q. What do you mean by Constant Comments? How constant use in Program?

Ans:

A constant comment is a note that we put into our source code. All comments are ignored by the compiler. Comments are used primarily to document the meaning

1.single line comment start with

//this is comment line

2.Multiple line comment

They start with /* and terminates with the Characters and end with */ as shown below:

We can use a comment to temporarily rmove a line of code.

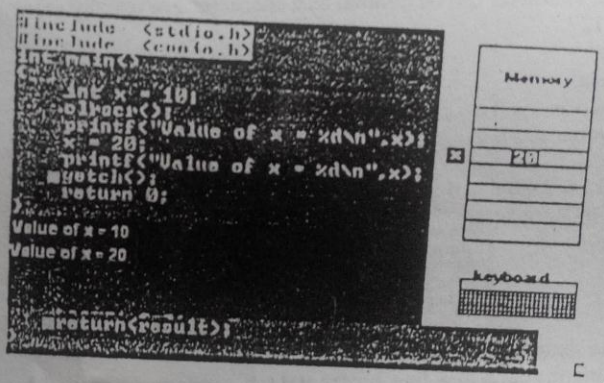
Q.What is a variable?
Ans:

A variable is the name storage

Q.Define variable and explain briefly wit example? ICE-2013,2015,APPE-2012
Ans:
Definition:- Variables are named memory locations that have a type, such as integer or character, which is inherited from their type.
Or

Avariable is a data name that may be used to store a data value. A varaible name can be chosen by the programmer in a meaningful way.
Example:

int *num*;
Here, *num* is a variable of integer type.



Any combination of letters, numbers, and underscore (_)
Case matters

- "sum" is different than "Sum"
- Cannot begin with a number
- usually, variables beginning with underscore
- are used only in special library routines
- Only first 31 characters are used

Legal

i
wordsPerSecond
_green

Illegal

10sdigit
done?
Double

There are two type of variable:

(a) Local Variables
(b) Global Variables

Local Variables: Variables that are declared inside a function or block are local variables. only accessible in a particular region
Global Variables:Global variables are defined outside of all the functions, usually on top of the program.A global variable can be accessed by any function or anywhere

Q.Define Datatype? CSE-
Ans:
All c compilers support five fundamental data typees these are as follows.
Those are:-

(a) Integer
(b) Floating point
(c) Character
(d) Double

(e) Void (valueless)

Data type	Meaning	Size (byte)	Minimal range
char	Character	1	-128 to 127
int	Integer	2	-32,768 to 32,768
float	Single precision real number	4	3.4E-38 to 3.4E+38
double	Double precision real number	8	1.7E-308 to 1.7E+308
void	Valueless	0	

* In C language '%d' is used for Integer.
* '%f' is used for floating point
* '%c' is used for character type data.

Integer Type: The variable of integer type are used to represent the integer type data . Integer type require two bytes of 16 bits of internal storage. Its range is -32768 ro 32767.

Character type: Character set are the set of alphabets, letters and some special characters that are valid in C language. character can be defined as a character are usually stored in 8 bytes of internal storage. Its range is -128 to 127.

Example:

Alphabets:

Uppercase: A B C ..... X Y Z

Lowercase: a b c ..... x y z

Digits:

0 1 2 3 4 5 6 8 9

Special Characters:

Special Characters in C language

, < > . _ ( ) : $: % [ ] # ?
' & ( ) " ^ ! * / | - \ ~ + ;

White space Characters:

blank space, new line, horizontal tab, carriage return and form feed

Floating Type: The variable of Floating type are used to represent the floating point type numbers.Integer type require 4 bytes of 32 bits of internal storage. Its range is -3.4 E+38 to 3.4 E-38.

Double Type: The variable of double types are also used to represent the real numbers. But its requirement is double to the floating type that is 8 bytes or 64bits its range is 1.7 E-308 to 1.7 E+308

Void type: The void types has no values . this is usually used to specify the type of functions. The type of a function is said to be void when it does not return any value to calling functions.

Another figure:

Type	Size (byte)	Minimal range
char	1	-127 to 127
unsigned char	1	0 to 255
signed char	1	-127 to +127
int	2 or 4	-32,767 to 32,767
unsigned int	2 or 4	0 to 65,535
signed int	2 or 4	-32,767 to 32,767
short int	2	-32,767 to 32,767
unsigned short int	2	0 to 65,535
signed short int	2	-32,767 to 32,767
long int	4	-2,147,483,647 to 2,147,483,647
signed long int	4	-2,147,483,647 to 2,147,483,647
unsigned long int	4	0 to 4,294,967,295
unsigned long int	4	0 to 4,294,967,295
float	4	3.4E-38 to 3.4E+38 with 6 digits of precision
double	8	1.7E-308 to 1.7E+308 with 10 digits of precision
long double	8	1.7E-308 to 1.7E+308 with 10 digits of precision

Example- # include<stdio.h>

include< conio.h>

```
Void main ( )
{
clrscr ( );
int i;          //integer type size of I is 2 byte
char c;         //character type  size of c is 1 byte
float d;        //Floating type  size of d is 8 byte

printf ("Enter your number :');
scanf ("%d", & i) ;
printf ("Enter our character :");
scanf ("%c",&c);
printf ("Enter your floating number :");
Scanf ("%f"; & d);
printf ("you Entered %d %c %f",I, c,d);
getch ( );
}
```

If anyone wants a floating value just like 4.333 then he has to define it in the program.

Example:-

```
#include <stdio.h>
#include <conio.h>
int main ( )
{
int a, b ;
float c ;
C= (a+b)/4 ;
Print("Enter your 1st number:");
Scanf("%d", & a);
Printf("Enter your 2nd number:");
Scanf("%d", & b);
Printf("in the result is %.2f",c);
Gethe ( );
```

Q.What is different between Float and double variable?
Ans:

Difference between float and double:

Generally the size of float(Single precision float data type) is 4 bytes and that of double(Double precision float data type) is 8 bytes. Floating point variables has a precision of 6 digits whereas the the precision of double is 14 digits.

Q.What is Sizeof operator ? and use of its give with an example?
Ans:

The sizeof operator It is a unary operator which is used in finding the size of data type, constant, arrays, structure etc. For example:

```
#include <stdio.h>
int main(){
  int a;
```

```
    float b;
    double c;
    char d;
    printf("Size of int=%d bytes\n",sizeof(a));
    printf("Size of float=%d bytes\n",sizeof(b));
    printf("Size of double=%d bytes\n",sizeof(c));
    printf("Size of char=%d byte\n",sizeof(d));
    return 0;
}
```

Output

```
Size of int=4 bytes
Size of float=4 bytes
Size of double=8 bytes
Size of char=1 byte
```

).Discuss I/O integers in C?

Ans:

```
#include<stdio.h>
int main()
{
  int c=5;
  printf("Number=%d",c);
  return 0;
}
```

Output

```
Number=5
```

iside quotation of printf() there, is a conversion format string "%d" (for integer). If this conversion format ring matches with remaining argument,i.e, *c* in this case, value of *c* is displayed.

```
#include<stdio.h>
int main()
{
  int c;
  printf("Enter a number\n");
  scanf("%d",&c);
  printf("Number=%d",c);
  return 0;
}
```

Output

```
Enter a number
4
Number=4
```

) of characters and ASCII code

```
#include <stdio.h>
int main(){
  char var1;
  printf("Enter character: ");
  scanf("%c",&var1);
  printf("You entered %c.",var1);
  return 0;
}
```

Output

```
Enter character: g
```

You entered g.

ASCII code

When character is typed in the above program, the character itself is not recorded a numeric value(ASCII value) is stored. And when we displayed that value by using "%c", that character is displayed.

```
#include <stdio.h>
int main(){
  char var1;
  printf("Enter character: ");
  scanf("%c",&var1);
  printf("You entered %c.\n",var1);
  printf("ASCII value of %d",var1);
  return 0;
}
Output
Enter character:
g
103
```

```
#include <stdio.h>
int main(){
  int var1=69;
  printf("Character of ASCII value 69: %c",var1);
  return 0;
}
Output
Character of ASCII value 69: E
```

Q.What is ASCII ?

Ans:

abbreviated from **American Standard Code for Information Interchange**,is a character-encoding scheme. ASCII codes represent text in computers, communications equipment, and other devices that use text. Text information is made of lots of individual units of information called *ASCII characters*

An ASCII character is often just a key stroke on the computer keyboard.

Examples of ASCII Characters :

An ASCII character can be:

- A lower case letter of the English alphabet.
- An upper case letter of the English alphabet.
- A single digit from zero to nine.
- A single space.
- A punctuation mark such as . or , or : or ; etc
- A math symbol such as + or - or / or (or) or = etc
- A special character such as # or $ or ^ or | etc

Consequently when text information is stored on or used by a computer, its ASCII characters must be converted into the form of binary numbers using a table of ASCII codes.

Normally you will convert ASCII to binary by first converting it to hexadecimal using the hexadecimal code table. Then, to get to binary.

Q; write a program which reads your name from the keyboard and outputs a listh of ASCII code, which represent your name. Exam:ACCE-2012,11,CSE,APPE

Ans:

```
#include<stdio.h>
int main()
{
int i;
```

```
char name[50];
printf("Enter your name: ");
gets(name);
printf("nCharactertASCII Code");
for(i=0;name[i]!=';i++)
printf("n%ctt%d",name[i],name[i]);
return 0;
}
```

Q.What is overflow and underflow?
Ans:

Overflow: *Overflow* is said to occur when the number that one gets as a result of some arithmetic operation on two *n*-bit binary numbers(signed or unsigned) is larger in magnitude than the largest number one can represent using *n*-bits. When adding and subtracting integers, using n-bit 2's complement representation, overflow may occur in 4 different situations, identify these.

or

When stack is full and yet another operation tries to PUSH data onto the stack then a 'stack overflow' occurs

Underflow:

Talking about underflow makes sense for floating-point numbers. For example using a 4-bit 2's complement representation for an exponent of a floating-point number, would enable us to represent numbers of the order of 10^{-16} to 10^{15}. If the number is less than 10^{-16} there would be no way to represent it, and hence it would lead to an "underflow". Underflow in floating-point arithmetic may also be thought of as overflow of the exponent.

or

If the stack is empty and yet a POP operation is attempted, this is called an 'stack underflow'.

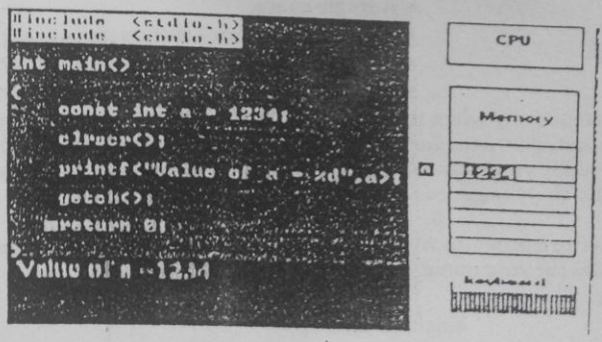
Q.What is constants?
Ans:

Constants in C refers to fixed value that do no t change during the execution of a program. the constant definitions come before the variable declarations.
-constant -identifier -constant.

There are 4 type of constant in C . they are

(a) Integer constant
(b) Floating constant
(c) Character constant
(d) String constant

Constants are data values that cannot be changed during the execution of a program.



Q.Discuss defferent technique of define constants? Exam-2011
Ans:

There are two technique in C to define constants:

1. Using #define preprocessor.
2. Using const keyword

1.The Define

#define identifier value

Here The #define is a Preprocessor Following is the form to use
#define preprocessor used to define a constant:

Example:
Here LENGTH, WIDTH are identifier name or vaiable name. we can these variable as a constant by defining with #define keyword.

```
#include <stdio.h>
#define LENGTH 10
#define WIDTH 5
#define NEWLINE '\n'
int main()
{
int area;
area = LENGTH * WIDTH;
 printf("value of area : %d", area);
 printf("%c", NEWLINE);
 return 0;
 }
```

2.The const Keyword

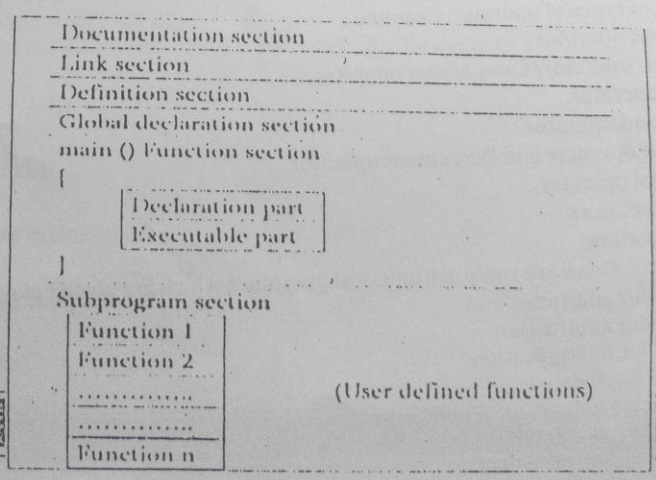
```
const type variable = value
```

The const is a Preprocessor Following is the form to use Const preprocessor used to define a constant:
Example:
;
Here LENGTH, WIDTH are identifier name or vaiable name. we can these variable as a constant by defining with const keyword

```
#include <stdio.h>
const int LENGTH = 10;
const int WIDTH = 5;
const char NEWLINE = '\n'
int main()
{
;
int area;
area = LENGTH * WIDTH;
printf("value of area : %d", area);
printf("%c", NEWLINE);
return 0;
}
```

Q.Discuss the basic structure of c?
Ans:

A C program may contain one or more sections. The main structral component of c program consists of one or more modules called functions. They are illustrated below



1. Documentation section : the author and other details, which the programmer would like to use later.
2. Link section : The link section provides instructions to the

compiler to link functions from the system library.

3. **Definition section :** The definition section defines all symbolic constants.
4. **Global declaration section :** There are some variables that are used in more than one function
5. **main () function section :** Every C program must have one main function section. This section contains two parts; declaration part and executable part
6. **Declaration part :** The declaration part declares all the variables used in the executable part.
7. **Executable part :** There is at least one statement in the executable part. These two parts must appear between the opening and closing braces..
8. **Subprogram section :** The subprogram section contains all the user-defined functions that are called in the main () function.

Q.Discuss programming input and output with example?

Ans: ANSI standard has defined many library functions for input and output in C language. Functions printf() and scanf() are the most commonly used to display out and take input respectively.

```
#include <stdio.h>     //This is needed to run printf() function.
int main()
{
   printf("C Programming"); //displays the content inside quotation
   return 0;
}
```

Output

C Programming

Explanation of How this program works

1. Every program starts from main() function.
2. printf() is a library function to display output which only works if #include<stdio.h>is included at the beginning.
3. Here, stdio.h is a header file (standard input output header file) and #include is command to paste the code from the header file when necessary. When compiler encounters printf() function and doesn't find stdio.h header file, compiler shows error.
4. Code return 0; indicates the end of program. You can ignore this statement but, it is good programming practice to use return 0;.

Q. Describe the different types of operator with suitable example?

Ans:

There are eight types of operator. Example :

1.Arithmatic operator
2.Relational operator/Comparison operator
3. Logical operator
4. Assignment operator
5.Increment operator and Decrement operator.
6.conditional operator.
7.Bitwies operators.
8.Special operator.

(a) **Arithmetic operator :-** There are some arithmetical operators which are given below :

i) "+" for addition
ii) "-" For subtraction
iii)"*" for multiplication
iv) "/" for division
v) "%" for remain or modulus division
vi)"--" for decrement
vii) "++" for increment

Relational operator :- The following relational operators are in C language :

i) ">" grater than
ii) "<" less than
iii) "<=" less than or equal
iv) ">=" grater than or equal
v) "==" equal
vi) "!=" not equal

Relational operators are also called on comparison operator.

(c) Logical operator :- This operator is used for formal logic. There are some logical operators are in C language. Those are:

i)"&&" AND
ii) "||" OR
ii) "!" NOT

(d)Assignment Operator:

or

Q.Replace +,-,/,* and = with equivalent statement using only ++,--,and +=,*=,/=%. Exam:ACCE-2012

(i) a=a-3; Ans:a-=2;
(ii) a=a*2 Ans: a*=2;
(iii) a=a/4 Ans: a/=4;
(iv)a=a%2 Ans:a%=2;
(v) b=b+(c+2) Ans: b+=c+2;
(vi)d=d*(n-5) Ans: d*=n-5;

(E)Conditional Operator:

or

Q. What is conditional operator. Give with an example?

Ans:

Conditional operators (?:)

Conditional operators are used in decision making in C programming, i.e, executes different statements according to test condition whether it is either true or false.

Syntax of conditional operators

conditional_expression?expression1:expression2

If the test condition is true, expression1 is returned and if false expression2 is returned.

Example of conditional operator

```
#include <stdio.h>
int main(){
  char feb;
  int days;
  printf("Enter I if the year is leap year otherwise enter 0: ");
  scanf("%c",&feb);
  days=(feb=='I')?29:28;
  /*If test condition (feb=='I') is true, days will be equal to 29. */
```

&	Bitwise AND
|	Bitwise OR
^	Bitwise exclusive OR
~	Bitwise complement
<<	Shift left

```
    /*If test condition (feb=='l') is false, days will be equal to 28. */
    printf("Number of days in February = %d",days);
    return 0;
}
```

Output

```
Enter l if the year is leap year otherwise enter n: l
Number of days in February = 29
```

(E)Bitwise Operator:

Q.Determine the output of the program.

```
Int amount,count;
Count=3;
Amount=2*++count;
Printf("count=%d\n",count,amount);
```

Ans: count=4;

Q. Are there any problems with performing mathematical operations on different variable types?

Ans:

C has three categories of built-in data types:
pointer types, integer types, and floating-point types.

Pointer types are the most restrictive in terms of the operations that can be performed on them. They are limited to - subtraction of two pointers, valid only when both pointers point to elements in the same array. The result is the same as subtracting the integer subscripts corresponding to the two pointers. + addition of a pointer and an integral type. The result is a pointer that points to the element which would be selected by that integer.

Floating-point types consist of the built-in types float, double, and long double. integer types consist of char, unsigned char, short, unsigned short, int, unsigned int, long, and unsigned long. All of these types can have the following arithmetic operations performed on them:

+ Addition
- Subtraction
* Multiplication
/ Division

Integer types also can have those four operations performed on them, as well as the following operations:

% Modulo or remainder of division
<<>> Shift right
& Bitwise AND operation
Bitwise OR operation
^ Bitwise exclusive OR operation
! Logical negative operation
~ Bitwise "one's complement" operation

Although C permits "mixed mode" expressions (an arithmetic expression involving different types), it actually converts the types to be the same type before performing the operations (except for the case of pointer arithmetic described previously). The process of automatic type conversion is called "operator promotion."

Q.Identify errors in the following program. After correction what output would you expect when you execute it?

```
#include<stdio.h>
#define PI 3.1416
int main()
 {
 int R,C;
 float perimeter;
 float area;
 C=PI;
 R=5;
 Perimeter=2.0*C*R;
 Area=C*R*R;
 printf("f","%d",&perimeter, &area);
 return 0;
 }
```

Ans:

There are many error in this program.

- We assigned Perimeter variable, but we declare perimeter. So Perimeter undeclared
- Area but we declared area. Area undeclared.
- In the printf function f error causes to print write %f and must under " " and area is floating type but print as a integer type.
- At output & perimeter &area these are logical error, print address of these variable.

After correction error's of this program :

```
#include<stdio.h>
#define PI 3.1416
int main()
 {
 int R=0,C=0;
 float perimeter=0;
 float area=0;
 C=PI;
 R=5;
 perimeter=2.0*C*R;
 area=C*R*R;
 printf("%f-%f", perimeter, area);
 return 0;
 }
```

output is: Perimeter=30 and area=75

Q. If a=15 and b=10 what will be the value of c in the following expression?

Ans:

C=++a-b Ans:6

C=b+++a; Ans:25

Q.Evaluate each of the following expression (The expression are to be evaluated independent of one naother another)

```
int i=3,j=4,k=2;
i. i--;j++
Ans:3,4
(ii).k+++*--i;
Ans:4
(iii) j+1/i+1
Ans:2
(iv) k+=k+i;j++
 k=5;j=4;
```

Q.Write a program to read the radius of a circle and compute its area and circumference. Exam:ACCE 2011,APPE,MSE,CSE;

Ans:

```
#include<stdio.h>
int main()
{
int r;
float cir, area;
const int PI=3.1427;
printf("\n ENTER THE RADIUS OF THE CIRCLE: ");
scanf("%d", &r);
cir = 2*PI*r;
area=PI*r*r;
printf("\n THE CIRCUMFERENCE OF CIRCLE IS= %f", cir);
printf("\n THE AREA OF CIRCLE IS= %f", area);
return 0;
}
```

OUTPUT:

ENTER THE RADIUS OF THE CIRCLE: 5
THE CIRCUMFERENCE OF CIRCLE IS= 31.427
THE AREA OF CIRCLE IS= 78.5675

Q. The following is a segment of a program:

```
int main()
{  int x=1,y=1,n=0;
if(n>0)
   x=x+1;
   y=y+1;
  printf("%d %d",x,y);
}
```

What will be the values of x and yif n assumes a vlalue of (a) 1 and (b) 0.

Ans:when n=1 ,x=2,y=2 when n=0, x=1,y=2;

Q.Assuming x=10, state whether the following logical expressions are true or false.

(i) x==10&&x>10&&!x Fasle
(ii) x==10||x>10||!x True
(iii) x==10&&x>10||!x False
(iv) x==10||x>10||!x False

Q. Analyze each of the following segmensts that follow and determine that follow and determine how many times the body of the each loop will execute.

(i).int main()

```
{
 int x=5,y=5;
 while(x<=y)
  {
   x=u/x;
  }
 return 0;
}
```

Ans: Error; Because U is undeclared

(ii). for(i=0;i<=5;i=i+2/3)
infiniete loop,value of i is always 0;

Q.Identify errors, if any , in each of the following array declaration statements, assuming theat ROW and COLUMN are declared as symbolic constant:

(iii)int score(100);

Ans: becasue score is not built in function , here absence declared score function. or its not an array;

(iv)float value[10,1.5] ;

Ans: if value as an array ,] must expect before , token operato. we cauld not assigned value of array in this procedure.

(v) float average[ROW],[COLUMN];

Ans: If Its a two dimenslonal array " , " token is not accepts between two dimensional array.

(vi)Int number[]={0,0,0,0,0},

Ans: its right procedur but , is not accept here.expect ; here.

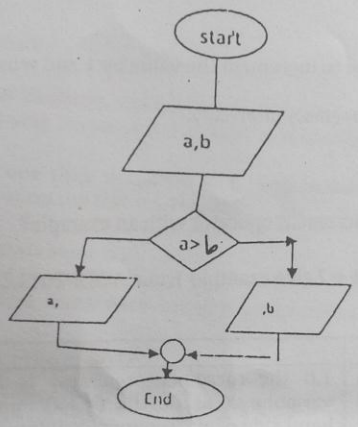
(vii)float Item[3][2]={0,1,2,3,4,5};

Ans: here no any physical erro but here logical error. that is item[][] is a two dimensional array , we assign here one dimensional value.

(viii) float result[10]=0;

Ans: invalid initializer valid is float result[10]={0};

Q.Write a flowchart for find largest number from 2 number?



Q.Write a program to find gretest number from two number?

Ans:

```
#Include <stdlo.h>
main()
{
int i,j,big;
scanf("%d%d",&i,&j);
);
if(i > j)
  {
  printf("biggest of two numbers is %d \n",i);
  }
 else
  {
    printf("biggest of two numbers is %d \n",j);
  }
}
```

Q.Describe Increment and decrement operator with example? Marks:1.25 Exam:ACCE-2014

Increment operators : Increment operators are increase the value of subsequent. value may be increase according to the programmer.

- Increment operator are two types as follows :
  1. Post Increment
  2. Pre Increment

Decrement operators (–)decrement operators decrease the value to one, two and so on.

- As like Increment operators, decrement operators are also two type as :
  1. Post decrement
  2. Pre decrement

Before we more discuss about increment and decrement operator, let understand some expression; that's are using in all operators.

Q .S++ or S = S+1, which can be recommended to Increment the value by 1 and why?

S++, as it is single machine instruction (INC) internally.

Q.Compare between pre increment and post increment operator with an example?

Or What is the difference between ++X and X++ ? give example Exam:ACCE-2011,CSE-ICE,APPE

++X	X++
1.if the increment operator is fixed in a variabl. X a prefix (++x) on the right hand side of an assignment statement.	1.If the increment operator is fixed in a variabbe. X as postfix (++x) on the right hand side of an assignment statement.
2.the compiler at first add one with the value of X and the result of adding in used in the statement.	2. the compiler at first uses the value of x in th statement and add one with the valu in the nex execution.
X=5 Y=++X In theis case value of Y and X are both 6.	X=5 Y=X++; In this case the value of Y is 5 and X is 6.

Q. What is a NULL statement?

Ans:

A null statement is no executable statements such as ; (semicolon).

Eg: int count = 0;

while(++count<=10);Above does nothing 10 times.