

GE6151 Computer Programming Question Bank

UNIT I – INTRODUCTION

PART A (2 MARKS)

1. Define computers?

A computer is a programmable machine or device that performs pre-defined or programmed computations or controls operations that are expressible in numerical or logical terms at high speed and with great accuracy.

(Or)

Computer is a fast operating electronic device, which automatically accepts and store input data, processes them and produces results under the direction of step by step program.

2. What are the basic operations of Computer?

- ✓ It accepts data or instructions by way of input.
- ✓ It stores data.
- ✓ It can process data as required by the user.
- ✓ It gives results in the form of output.
- ✓ It controls all operations inside a computer.

3. What are the characteristics of computers? (JAN 2009)

- ✓ Speed
- ✓ Accuracy.
- ✓ Automation.
- ✓ Endurance.
- ✓ Versatility.
- ✓ Storage.
- ✓ Cost Reduction.

GE6151 Computer Programming Question Bank

4. How will you classify computer systems? (JAN2009)

Based on physical size, performance and application areas, we can generally divide computers into four major categories:

- ✓ Micro computer
- Mini Computer
- ✓ Mainframe computer and
- ✓ Super Computer

5. Expand ENIAC, ABC, EDVAC, EDSAC and UNIVAC. (JAN2010)

ENIAC – Electronic Numerical Integrator and Calculator.

ABC – Atanasoff and Berry Computer.

EDVAC – Electronic Discrete Variable Automatic Calculator.

EDSAC – Electronic Delay Storage Automatic Calculator.

UNIVAC – UNiversal Automatic Computer.

6. What are the components of the computer systems?

Basic components of the computer system are Input Unit, Central Processing Unit, Secondary Storage Unit and Output Unit.

7. What is an ALU?

Arithmetic logic unit, the part of a computer that performs all arithmetic computations, such as addition and multiplication, and all logical operations such as comparison operations. The ALU is one component of the CPU (central processing unit).

8. What are the functions in the input unit?

An input device is a device that is used to input data or information into a computer. Some examples of input devices include:

- ✓ Keyboards, mouse.
- ✓ Light Pen.
- ✓ Digitizer.
- ✓ Touchpad.

- ✓ Speakers.
- ✓ Printer.
- ✓ Headphone

GE6151 Computer Programming Question Bank

9. What are the functions in the output unit?

In computers, a unit which delivers information from the computer to an external device or from internal storage to external storage.

10. Define Mainframe computer?

Mainframes are computers used mainly by large organizations for critical applications, typically bulk data processing such as census, industry and consumer statistics, enterprise resource planning, and financial processing. The term probably had originated from the early mainframes, as they were housed in enormous, room-sized metal boxes or frames.

11. Define super computer?

The fastest type of computer. Supercomputers are very expensive and are employed for specialized applications that require immense amounts of mathematical calculations. For example, weather forecasting requires a supercomputer. Other uses of supercomputers include animated graphics, fluid dynamic calculations, nuclear energy research, and petroleum exploration.

12. What is a volatile and non-volatile memory?

Volatile memory: also known as volatile storage is computer memory that requires power to maintain the stored information, unlike non-volatile memory which does not require a maintained power supply. It has been less popularly known as temporary memory.

Non-volatile memory: nonvolatile memory, NVM or non-volatile storage, is computer memory that can retain the stored information even when not powered.

13. Write the binary and octal equivalent of hexadecimal number 7BD? (APR2009)

Binary Equivalent of 7BD = (0111 1011 1101)₂

Octal Equivalent of 7BD = (011 110 111 101) = (3675)₈

14. Convert binary number 100110 into its octal equivalent? (JAN2009)

Octal equivalent of 100110 = (100 110) = (46)₈

15. What are registers? (JAN2009)

A special, high-speed storage area within the CPU. All data must be represented in a register before it can be processed.

GE6151 Computer Programming Question Bank

For example, if two numbers are to be multiplied, both numbers must be in registers, and the result is also placed in a register.

16. Differentiate analog and digital computers? (JAN2010)

Analog Computer Digital Computer

- ✓ Process measured data Process discrete data
- ✓ Analog computers are not precise Digital computers are more precise
- ✓ Processing speed is low. Processing speed is high.
- ✓ Less accuracy. More accuracy.

17. Find the decimal equivalent of hexadecimal number 4D.C8 (JAN2010)

$$4D.C8 = 4 \times 16^1 + 13 \times 16^0 + 12 \times 16^{-1} + 8 \times 16^{-2}$$
$$= 64 + 13 + 0.75 + 0.03125 = (77.78)_{10}$$

18. Convert hexadecimal number into binary equivalent of EBC (JAN2010)

Binary equivalent of EBC = (1110 1011 1100)₂

19. What is algorithm? (JAN2009)

Algorithm means the logic of a program. It is a step-by-step description of how to arrive at a solution of a given problem.

20. What are the steps to solve the problem in a computer system?

- ✓ Problem must be analyzed thoroughly.
- ✓ Solution method is broken down into a sequence of small tasks.
- ✓ Based on this analysis, an algorithm must be prepared to solve the problem.
- ✓ The algorithm is expressed in a precise notation. This notation is known as “Computer Program”.
- ✓ The Computer program is fed to the computer.
- ✓ The instruction in the program executes one after another and outputs the expected result.

21. What are the characteristics of an algorithm?

- ✓ In algorithms each and every instruction should be precise.
 - ✓ In algorithms each and every instruction should be unambiguous.
-

GE6151 Computer Programming Question Bank

- ✓ The instructions in an algorithm should not be repeated infinitely.
- ✓ Ensure that the algorithm will ultimately terminate.
- ✓ The algorithm should be written in sequence.
- ✓ It looks like normal English.

22. What is Flowchart?

A Flowchart is a pictorial representation of an algorithm. It is often used by programmer as a program planning tool for organizing a sequence of step necessary to solve a problem by a computer.

23. What is the need of Flowchart symbols?

Each symbol of different shapes denotes different shapes denote different types of instructions. The program logic through flowcharts is made easier through the use of symbol that has standardized planning.

24. What is pseudo code?

“Pseudo” means imitation of false and “code” refers to the instruction written in the programming language. Pseudo code is programming analysis tool that is used for planning program logic.

UNIT-II

2 MARKS

1. Define Compilation process.

Compilation refers to the processing of source code files (.c, .cc, or .cpp) and the creation of an 'object' file. This step doesn't create anything the user can actually run. Instead, the compiler merely produces the machine language instructions that correspond to the source code file that was compiled.

2. What do you meant by linking?

Linking refers to the creation of a single executable file from multiple object files. In this step, it is common that the linker will complain about undefined functions (commonly, main itself). During compilation, if the compiler could not find the definition for a particular function, it would just assume that the function was defined in another file. If this isn't the case, there's no way the compiler would know it doesn't look at the contents of more than one file at a time. The linker, on the other hand, may look at multiple files and try to find references for the functions that weren't mentioned.

3. Define Constants in C. Mention the types.

The constants refer to fixed values that the program may not alter during its execution. These fixed values are also called **literals**.

Constants can be of any of the basic data types like *an integer constant, a floating constant, a character constant, or a string literal*. There are also enumeration constants as well.

The **constants** are treated just like regular variables except that their values cannot be modified after their definition.

4. What are the different data types available in 'C'?

There are four basic data types available in 'C'.

1. int
2. float
3. char

GE6151 Computer Programming Question Bank

4. double

5. What is meant by Enumerated data type.

Enumerated data is a user defined data type in C language.

Enumerated data type variables can only assume values which have been previously declared.

Example :

```
enum month { jan = 1, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec };
```

6. What are Keywords?

Keywords are certain reserved words that have standard and pre-defined meaning in 'C'. These keywords can be used only for their intended purpose.

7. What do you mean by variables in 'C'?

- ✓ A variable is a data name used for storing a data value.
- ✓ Can be assigned different values at different times during program execution.
- ✓ Can be chosen by programmer in a meaningful way so as to reflect its function in the program.
- ✓ Some examples are: Sum

8. Difference between Local and Global variable in C.

Local

These variables only exist inside the specific function that creates them. They are unknown to other functions and to the main program. As such, they are normally implemented using a stack. Local variables cease to exist once the function that created them is completed. They are recreated each time a function is executed or called.

GE6151 Computer Programming Question Bank

Global

These variables can be accessed (ie known) by any function comprising the program. They are implemented by associating memory locations with variable names. They do not get recreated if the function is recalled.

9. What are Operators? Mention their types in C.

An operator is a symbol that tells the compiler to perform specific mathematical or logical manipulations. C language is rich in built-in operators and provides following type of operators:

- Arithmetic Operators
- Relational Operators
- Logical Operators
- Bitwise Operators
- Assignment Operators
- Misc Operators

10. What is the difference between '=' and '==' operator?

Where = is an assignment operator and == is a relational operator.

Example:

while (i=5) is an infinite loop because it is a non zero value and while (i==5) is true only when i=5.

11. What is type casting?

Type casting is the process of converting the value of an expression to a particular data type.

Example:

```
int x,y;
```

```
c = (float) x/y; where a and y are defined as integers. Then the result of x/y is converted into float.
```

12. What is the difference between ++a and a++?

++a means do the increment before the operation (pre increment)

a++ means do the increment after the operation (post increment)

Example:

GE6151 Computer Programming Question Bank

```
a=5;
x=a++; /* assign x=5*/ y=a;
/*now y assigns y=6*/
x=++a; /*assigns x=7*/
```

13.Distinguish between while..do and do..while statement in C. (JAN 2009)

While	DO..while
(i) Executes the statements within the while block if only the condition is true.	(i) Executes the statements within the while block at least once.
(ii) The condition is checked at the starting of the loop	(ii) The condition is checked at the end of the loop

14.Mention the various Decisions making statement available in C.

Statement	Description
if statement	An if statement consists of a boolean expression followed by one or more statements.
if...else statement	An if statement can be followed by an optional else statement , which executes when the boolean expression is false.
nested if statements	You can use one if or else if statement inside another if or else if statement(s).
switch statement	A switch statement allows a variable to be tested for equality against a list of values.
nested switch statements	You can use one switch statement inside another switch statement(s).

GE6151 Computer Programming Question Bank

15. What do you mean by conditional or ternary operator?

? : If Condition is true ? Then value X : Otherwise value Y

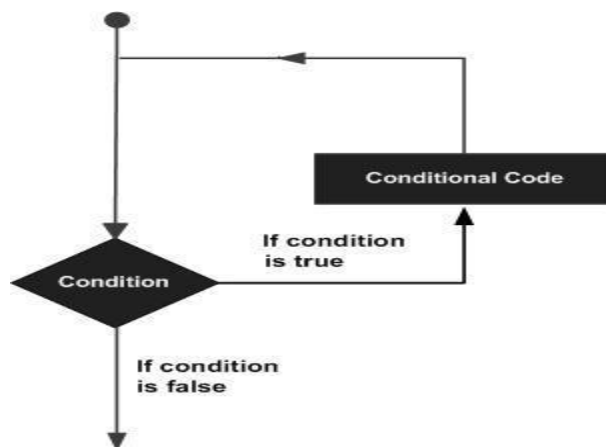
16. What is the use of sizeof() operator in C.

Sizeof operator is used to return the size of an variable.

Example : sizeof(a), Where a integer, will return 4.

17. Define Looping in C .

A loop statement allows us to execute a statement or group of statements multiple times and following is the general form of a loop statement in most of the programming languages:



18. What are the types of looping statements available in C

C programming language provides following types of loop to handle looping requirements. Click the following links to check their detail.

Loop Type

Description

while loop

Repeats a statement or group of statements while a given condition is true. It tests the condition before executing the loop body.

GE6151 Computer Programming Question Bank

for loop	Execute a sequence of statements multiple times and abbreviates the code that manages the loop variable.
do...while loop	Like a while statement, except that it tests the condition at the end of the loop body
nested loops	You can use one or more loop inside any another while, for or do..while loop.

19. What are the types of I/O statements available in 'C'?

There are two types of I/O statements available in 'C'.

- ✓ Formatted I/O Statements
- ✓ Unformatted I/O Statements

20. Write short notes about main () function in 'C' program. (MAY 2009)

- ✓ Every C program must have main () function.
- ✓ All functions in C, has to end with '(') parenthesis.
- ✓ It is a starting point of all 'C' programs.
- ✓ The program execution starts from the opening brace '{' and ends with closing brace '}', within which executable part of the program exists.

21. Define delimiters in 'C'.

Delimiters Use

: Colon

; Semicolon

() Parenthesis

[] Square Bracket

{ } Curly Brace

Hash

, Comma

GE6151 Computer Programming Question Bank

22. Why header files are included in 'C' programming?

· This section is used to include the function definitions used in the program. · Each header file has 'h' extension and include using '#include' directive at the beginning of a program.

23. What is the output of the programs given below?

```
main()
{
float a;
int x=6, y=4;
a=x\y;
printf("Value of a=%f", a);
}
```

Output:

1. 1.500000

```
main()
{
float a;
int x=6, y=4;
a=(float) x\y;
printf("Value of a=%f",a);
}
```

Output:

24. What is the difference between scanf() and gets() function?

In scanf() when there is a blank was typed, the scanf() assumes that it is an end. gets() assumes the enter key as end. That is gets() gets a new line (\n) terminated string of characters from the keyboard and replaces the '\n' with '\0'.

25. What are the Escape Sequences present in 'C'

\n - New Line

\b - Backspace

\t - Form feed \'

- Single quote \\'

- Backspace \b -

Tab

\r - Carriage return

\a - Alert

\\" - Double quotes

GE6151 Computer Programming Question Bank

26.What is the difference between if and while statement?

If	While
(i) It is a conditional statement	(i) It is a loop control statement
(ii) If the condition is true, it executes some statements.	(ii) Executes the statements within the while block if the condition is true.
(iii) If the condition is false then it stops the execution the statements.	(iii) If the condition is false the control is transferred to the next statement of the loop.

27.Differentiate between formatted and unformatted you input and output functions?

Formatted I/P functions:

These functions allow us to supply the input in a fixed format and let us obtain the output in the specified form. Formatted output converts the internal binary representation of the data to ASCII characters which are written to the output file.

Unformatted I/O functions:

There are several standard library functions available under this category-those that can deal with a string of characters. Unformatted Input/Output is the most basic form of input/output. Unformatted input/output transfers the internal binary representation of the data directly between memory and the file.

GE6151 Computer Programming Question Bank

UNIT –III

2 MARKS

1. What is an array?

An array is a group of similar data types stored under a common name. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

Example:

```
int a[10];
```

Here a[10] is an array with 10 values.

2. What are the main elements of an array declaration?

- ✓ Array name
- ✓ Type and
- ✓ Size

3. How to initialize an array?

You can initialize array in C either one by one or using a single statement as follows:

```
double balance[5] = {1000.0, 2.0, 3.4, 17.0, 50.0};
```

The number of values between braces { } cannot be larger than the number of elements that we declare for the array between square brackets []. Following is an example to assign a single element of the array:

4. Why is it necessary to give the size of an array in an array declaration?

When an array is declared, the compiler allocates a base address and reserves enough space in the memory for all the elements of the array. The size is required to allocate the required space. Thus, the size must be mentioned.

GE6151 Computer Programming Question Bank

5. What is the difference between an array and pointer?

Difference between arrays and pointers are as follows.

Array	Pointer
Array allocates space automatically	1.Pointer is explicitly assigned to point to an allocated space.
2.It cannot be resized.	2.It can be resized using realloc ().
3.It cannot be reassigned.	3.Pointers can be reassigned.
4.Size of(array name) gives the number of bytes occupied by the array.	4.Sezeof(pointer name) returns the number of bytes used to store the pointer variable.

6. List the characteristics of Arrays.

All elements of an array share the same name, and they are distinguished from one another with help of an element number. Any particular element of an array can be modified separately without disturbing other elements.

7. What are the types of Arrays?

- 1.One-Dimensional Array
2. Two-Dimensional Array
3. Multi-Dimensional Array

8. Define Strings.

Strings:

The group of characters, digit and symbols enclosed within quotes is called as String (or) character Arrays. Strings are always terminated with '\0' (NULL) character. The compiler automatically adds '\0' at the end of the strings.

Example:

```
char name[]={ 'C', 'O', 'L', 'L', 'E', 'G', 'E', 'E', '\0' };
```

GE6151 Computer Programming Question Bank

9. Mention the various String Manipulation Functions in C.

S.N.	Function & Purpose
1	strcpy(s1, s2); Copies string s2 into string s1.
2	strcat(s1, s2); Concatenates string s2 onto the end of string s1.
3	strlen(s1); Returns the length of string s1.
4	strcmp(s1, s2); Returns 0 if s1 and s2 are the same; less than 0 if s1<s2; greater than 0 if s1>s2.
5	strchr(s1, ch); Returns a pointer to the first occurrence of character ch in string s1.
6	strstr(s1, s2); Returns a pointer to the first occurrence of string s2 in string s1.

10. What is the use of atoi() function?

C allows us to manipulate characters the same way we do with numbers. Whenever a character constant or character variable is used in an expression, it is automatically converted into integer value by the system.

For eg, if the machine uses the ASCII representation, then,

```
x = 'a'; printf("%d \n",x);
```

will display the number 97 on the screen.

The C library supports a function that converts a string of digits into their integer values. The function takes the form

```
x = atoi(string)
```

11. What is the use of 'typedef'?

It is used to create a new data using the existing type.

GE6151 Computer Programming Question Bank

Syntax: typedef data type name;

Example:

```
typedef int hours; /* Now, hours can be used as new datatype */
```

12. What is meant by Sorting?

Sorting refers to ordering data in an increasing or decreasing fashion according to some linear relationship among the data items. Sorting can be done on names, numbers and records.

13. What are the types of sorting available in C?

Insertion sort.

Merge Sort.

Quick Sort.

Radix Sort.

Heap Sort

Selection sort

Bubble sort

14. Define Heap Sort.

A sorting algorithm that works by first organizing the data to be sorted into a special type of binary tree called a heap. The heap itself has, by definition, the largest value at the top of the tree, so the heap sort algorithm must also reverse the order. It does this with the following steps:

1. Remove the topmost item (the largest) and replace it with the rightmost leaf. The topmost item is stored in an array.
2. Re-establish the heap.
3. Repeat steps 1 and 2 until there are no more items left in the heap.

GE6151 Computer Programming Question Bank

15. Define Bubble sort.

A simple but popular sorting algorithm. Bubble sorting is used frequently as a programming exercise because it is relatively easy to understand. It is not, however, particularly efficient. Other sorting algorithms, such as heap sorts, merge sorts and quick sorts, are used more often in real applications.

16. Define Searching.

Searching for data is one of the fundamental fields of computing. Often, the difference between a fast program and a slow one is the use of a good algorithm for the data set. Naturally, the use of a hash table or binary search tree will result in more efficient searching, but more often than not an array or linked list will be used. It is necessary to understand good ways of searching data structures not designed to support efficient search.

17. Mention the various types of searching techniques in C

- ✓ Linear search
- ✓ Binary search

18. What is linear search?

In Linear Search the list is searched sequentially and the position is returned if the key element to be searched is available in the list, otherwise -1 is returned. The search in Linear Search starts at the beginning of an array and move to the end, testing for a match at each item.

19. What is Binary search?

A binary search, also called a dichotomizing search, is a digital scheme for locating a specific object in a large set. Each object in the set is given a key. The number of keys is always a power of 2. If there are 32 items in a list, for example, they might be numbered 0 through 31 (binary 00000 through 11111). If there are, say, only 29 items, they can be numbered 0 through 28 (binary 00000 through 11100), with the numbers 29 through 31 (binary 11101, 11110, and 11111) as dummy keys.

GE6151 Computer Programming Question Bank

UNIT IV 2 MARKS

1. What are functions in C?

A function is a group of statements that together perform a task. Every C program has at least one function which is **main()**, and all the most trivial programs can define additional functions.

2. How will define a function in C?

Defining a Function:

The general form of a function definition in C programming language is as follows:

```
return_type function_name( parameter list )  
{  
    body of the function  
}
```

A function definition in C programming language consists of a *function header* and a *function body*. Here are all the of a function:

- ✓ Return Type
- ✓ Function Name
- ✓ Parameters
- ✓ Function Body

3. What are the steps in writing a function in a program?

a) Function Declaration (Prototype declaration):

Every user-defined functions has to be declared before the main().

b) Function Callings:

The user-defined functions can be called inside any functions like main(),user-defined function, etc.

c) Function Definition:

4. What is the purpose of the function main()? (MAY 2009)

GE6151 Computer Programming Question Bank

The function main () invokes other functions within it. It is the first function to be called when the program starts execution.

5. Is it better to use a macro or a function?

Macros are more efficient (and faster) than function, because their corresponding code is inserted directly at the point where the macro is called. There is no overhead involved in using a macro like there is in placing a call to a function.

However, macros are generally small and cannot handle large, complex coding constructs. In cases where large, complex constructs are to handled, functions are more suited, additionally; macros are expanded inline, which means that the code is replicated for each occurrence of a macro.

6. Distinguish between Call by value Call by reference.

Call by value	Call by reference.
a) In call by value, the value of actual arguments is passed to the formal arguments and the operation is done on formal arguments.	a) In call by reference, the address of actual argument values is passed to formal argument values.
b) Formal arguments values are photocopies of actual arguments values.	b) Formal arguments values are pointers to the actual argument values.
c) Changes made in formal arguments valued do not affect the actual arguments values.	c) Since Address is passed, the changes made in the both arguments values are permanent

7. What is meant by Recursive function?

If a function calls itself again and again, then that function is called Recursive function.

Example:

GE6151 Computer Programming Question Bank

```
void recursion()
{
    recursion(); /* function calls itself */
}

int main()
{
    recursion();
}
```

8. Write a C program for factorial using recursion function.

Following is an example which calculates factorial for a given number using a recursive function:

```
#include <stdio.h>

int factorial(unsigned int i)
{
    if(i <= 1)
    {
        return 1;
    }
    return i * factorial(i - 1);
}

int main()
{
    int i = 15;
    printf("Factorial of %d is %d\n", i,
        factorial(i)); return 0;
}
```

When the above code is compiled and executed, it produces the following result:

```
Factorial of 15 is 2004310016
```

9. What is dynamic memory allocation?

Allocating the memory at run time is called as dynamic memory allocation.

10. What are the various dynamic memory allocation functions?

malloc() - Used to allocate blocks of memory in required size of bytes.

free() - Used to release previously allocated memory space.

GE6151 Computer Programming Question Bank

calloc() - Used to allocate memory space for an array of elements.

realloc() - Used to modify the size of the previously allocated memory space.

11. What is a Pointer? How a variable is declared to the pointer? (MAY 2009)

Pointer is a variable which holds the address of another variable.

Pointer Declaration:

```
datatype *variable-name;
```

Example:

```
int *x, c=5;
```

```
x=&a;
```

12. What are the uses of Pointers?

Pointers are used to return more than one value to the function

- Pointers are more efficient in handling the data in arrays
- Pointers reduce the length and complexity of the program
- They increase the execution speed
- The pointers save data storage space in memory

13. What is the output of the program?

```
main()                                junk(int i, int j)
{                                       {
int i=5;j=2;                            i=i*j;
junk(i,j);                               j=i*j;
printf("\n %d %d",i,j);                 }
}
```

Output:

1. 2

2.

GE6151 Computer Programming Question Bank

14. What are * and & operators means?

'*' operator means 'value at the address'

'&' operator means 'address of'

15. What is dangling pointer?

In C, a pointer may be used to hold the address of dynamically allocated memory. After this memory is freed with the free() function, the pointer itself will still contain the address of the released block. This is referred to as a dangling pointer. Using the pointer in this state is a serious programming error. Pointer should be assigned NULL after freeing memory to avoid this bug.

16. What is meant by Recursive function?

If a function calls itself again and again, then that function is called Recursive function.

Example:

```
void recursion()
{
    recursion(); /* function calls itself */
}

int main()
{
    recursion();
}
```

17. Write a C program for factorial using recursion function.

Following is an example which calculates factorial for a given number using a recursive function:

```
#include <stdio.h>

int factorial(unsigned int i)
{
    if(i <= 1)
    {
        return 1;
    }
}
```

GE6151 Computer Programming Question Bank

```
    return i * factorial(i - 1);
}
int main()
{
    int i = 15;
    printf("Factorial of %d is %d\n", i,
        factorial(i)); return 0;
}
```

When the above code is compiled and executed, it produces the following result:

```
Factorial of 15 is 2004310016
```

18. What is dynamic memory allocation?

Allocating the memory at run time is called as dynamic memory allocation.

19. What are the various dynamic memory allocation functions?

malloc() - Used to allocate blocks of memory in required size of bytes.

free() - Used to release previously allocated memory space.

calloc() - Used to allocate memory space for an array of elements.

realloc() - Used to modify the size of the previously allocated memory space.

GE6151 Computer Programming Question Bank

UNIT V 2 MARKS

1. Compare arrays and structures.

Comparison of arrays and structures is as follows.

Arrays	Structures
An array is a collection of data items of same data type. Arrays can only be declared.	A structure is a collection of data items of different data types. Structures can be declared and defined.
There is no keyword for arrays.	The keyword for structures is struct.
An array name represents the address of the starting element.	A structure name is known as tag. It is a Shorthand notation of the declaration.
An array cannot have bit fields.	A structure may contain bit fields.

2. Compare structures and unions.

Structure	Union
Every member has its own memory.	All members use the same memory.
The keyword used is struct.	The keyword used is union.
All members occupy separate memory location, hence different interpretations of the same memory location are not possible. Consumes more space compared to union.	Different interpretations for the same memory location are possible. Conservation of memory is possible

GE6151 Computer Programming Question Bank

3. Define Structure in C.

C Structure is a collection of different data types which are grouped together and each element in a C structure is called member.

If you want to access structure members in C, structure variable should be declared.

Many structure variables can be declared for same structure and memory will be allocated for each separately.

It is a best practice to initialize a structure to null while declaring, if we don't assign any values to structure members.

4. What you meant by structure definition?

A structure type is usually defined near to the start of a file using a typedef statement. typedef defines and names a new type, allowing its use throughout the program. typedefs usually occur just after the #define and #include statements in a file.

Here is an example structure definition.

```
typedef struct { char
    name[64];
    char course[128];
    int age;
    int year;
} student;
```

This defines a new type student variables of type student can be declared as follows.

```
student st_rec;
```

5. How to Declare a members in Structure?

A **struct** in C programming language is a structured (record) type^[1] that aggregates a fixed set of labeled objects, possibly of different types, into a single object. The syntax for a struct declaration in C is:

GE6151 Computer Programming Question Bank

```
struct tag_name
{
    type attribute;
    type attribute2;
    /* ... */
};
```

6. What is meant by Union in C.?

A **union** is a special data type available in C that enables you to store different data types in the same memory location. You can define a union with many members, but only one member can contain a value at any given time. Unions provide an efficient way of using the same memory location for multi-purpose.

7. How to define a union in C.

To define a union, you must use the **union** statement in very similar was as you did while defining structure. The union statement defines a new data type, with more than one member for your program.

The format of the union statement is as follows:

```
union [union tag]
{
    member definition;
    member definition;
    ...
    member definition;
} [one or more union variables];
```

8. How can you access the members of the Union?

To access any member of a union, we use the **member access operator** (.). The member access operator is coded as a period between the union variable name and the union member that we wish to access. You would use **union** keyword to define variables of union type.

GE6151 Computer Programming Question Bank

9. What are the pre-processor directives?

Macro Inclusion

Conditional Inclusion

File Inclusion

10. What are storage classes?

A storage class defines the scope (visibility) and life time of variables and/or functions within a C Program.

11. What are the storage classes available in C?

There are following storage classes which can be used in a C Program

auto

register

static

extern

12. What is register storage in storage class?

Register is used to define local variables that should be stored in a register instead of RAM. This means that the variable has a maximum size equal to the register size (usually one word) and cant have the unary '&' operator applied to it (as it does not have a memory location).

```
{  
register int Miles;  
}
```

13. What is static storage class?

GE6151 Computer Programming Question Bank

Static is the default storage class for global variables. The two variables below (count and road) both have a static storage class.

```
static int
Count; int Road;
{
    printf("%d\n", Road);
}
```

14. Define Auto storage class in C.

auto is the default storage class for all local variables.

```
{
int Count; auto
int Month;
}
```

The example above defines two variables with the same storage class. auto can only be used within functions, i.e. local variables.

15. Define pre-processor in C.

The C Preprocessor is not part of the compiler, but is a separate step in the compilation process. In simplistic terms, a C Preprocessor is just a text substitution tool. We'll refer to the C Preprocessor as the CPP.

Example:

#define Substitutes a preprocessor macro
#include Inserts a particular header from another file

GE6151 Computer Programming Question Bank

16. Define Macro in C.

A macro definition is independent of block structure, and is in effect from the #define directive that defines it until either a corresponding #undef directive or the end of the compilation unit is encountered.

Its format is: #define identifier replacement

Example:

```
#define TABLE_SIZE 100
int table1[TABLE_SIZE];
int table2[TABLE_SIZE];
```

17. What are conditional Inclusions in Preprocessor Directive?

Conditional inclusions (#ifdef, #ifndef, #if, #endif, #else and #elif)

These directives allow including or discarding part of the code of a program if a certain condition is met. #ifdef allows a section of a program to be compiled only if the macro that is specified as the parameter has been defined, no matter which its value is.

For example:

```
1 #ifdef TABLE_SIZE
2 int table[TABLE_SIZE];
3 #endif
```

18. What you meant by Source file Inclusion in Preprocessor directive?

Source file inclusion (#include)

This directive has also been used assiduously in other sections of this tutorial. When the preprocessor finds an #include directive it replaces it by the entire content of the specified file. There are two ways to specify a file to be included:

```
1 #include "file"
2 #include <file>
```

GE6151 Computer Programming Question Bank

19. What is Line control?

Line control (#line)

When we compile a program and some error happens during the compiling process, the compiler shows an error message with references to the name of the file where the error happened and a line number, so it is easier to find the code generating the error.

The #line directive allows us to control both things, the line numbers within the code files as well as the file name that we want that appears when an error takes place. Its format is:

```
#line number "filename"
```

Where number is the new line number that will be assigned to the next code line. The line numbers of successive lines will be increased one by one from this point on.

