



SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

COIMBATORE – 35

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (UG & PG)



First Year, 1st Semester

2 Marks Question and Answer

Subject Code & Name: 23CST101&Problem solving and C Programming

UNIT - 1

INTRODUCTION TO PROBLEM SOLVING TECHNIQUES

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?

- Input
- Process
- Storing
- Controlling and
- Output.

3. What are the characteristics of computer?

- Speed
- Accuracy
- Automation
- Endurance

- Versatility
- Storage and
- Cost reduction

4. Define data and information.

Data: Raw material for information processing

Information: The processed data is called information.

5. List out the operations of CPU.

CPU performs all the calculations, co-ordinates all the units of the computer, interprets the instruction and stores the data's.

6. Define system software.

A set of program that governs the operation of a computer system and makes the hardware works. It controls the internal operations of the computer.

7. What is software?

Software is a set of instruction or program, which are useful for performing a task.

8. Define operating system?

A collection of program used to control the entire operation of the computer. It co-ordinates the hardware and software.

9. What is application software?

Software which is used to solve a specific task is called application software.

10. What is meant by Information technology?

IT refers to the creation, gathering, processing, storage and delivery of information.

11. What is Algorithm?

Algorithm means the logic of a program. It is a step by step description of a program.

12. List out the way how algorithm may be represented?

- Normal English
- Flow chart
- Pseudocode
- Decision table

13. Define compiler.

It is a program used to convert the high level language program into machine language.

14. Define Assembler.

It is a program used to convert the assembly language program into machine language.

15. Define Interpreter.

It is a program used to convert the high level language program into machine language line by line.

16. What are the types of programming language?

- Machine language
- Assembly language
- High level language

17. Define flow chart.

Flow chart is the pictorial representation of a program.

18. List out the basic design structure?

- Sequence structure
- Selection structure
- Loop structure

PROBLEM SOLVING AND C PROGRAMMING

19. List out the any three advantages of flowchart.

- Better communication
- Effective synthesis
- Effective analysis

20. What is Data and Information?

Data - Data is the fact or raw material for the information processing.

Information – The processed data is called information.

21. Give the applications computer?

Word Processing

Internet

Desktop publishing

Digital video or audio composition

Mathematical Calculation

Robotics

Weather analysis

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

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

1. Micro computer
2. Mini Computer
3. Mainframe computer and
4. Super Computer

23. Specify the Electronic components used for different computer generations.

Generations	Electronic Components
I	Generation Vacuum tubes
II	Generation Transistors
III	Generation Integrated Circuits
IV	Generation Microprocessors
V	Generation Artificial Intelligence

24. What are the languages used in computer generations.

Generations	Languages used
I	Generation Machine Language
II	Generation Assemble Language, Mnemonics

PROBLEM SOLVING AND C PROGRAMMING

III	Generation High Level Language, BASIC, PASCAL, COBOL,
IV	Generation 4GL
V	Generation Artificial Intelligence

25. 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.

26. 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).

27. What is a CPU?

The CPU (central processing unit) is the part of a computer that controls the interpretation and execution of instructions. Generally, the CPU is a single microchip.

28. What is a Statement?

Statement is a single action in a computer. In a computer, statements might include some of the following actions

1. **input data**-information given to the program
2. **process data**-perform operation on a given input
3. **output data**-processed result

29. 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.

30. Define Mini computers?

A mini computer is a multi-user or time-sharing system. It is used for medium scale data processing such as Bank account processing, Payroll processing etc., Mini computer processes greater storage capacity and larger memories as compared to micro computer.

31. 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, control unit and ALU. Today, the CPUs of almost all computers are contained on a single chip. For example, weather forecasting requires a supercomputer. Other uses of supercomputers include animated graphics, fluid dynamic calculations, nuclear energy research, and petroleum exploration.

32. 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.

33. What is a primary memory?

The primary memory or the main memory is part of the main computer system. The processor or the CPU directly stores and retrieves information from it. This memory is accessed by CPU, in random fashion. That means any location of this memory can be accessed by the CPU to either read information from it, or to store information in it.

34. What is a secondary memory?

The secondary memory is much slower and also less costly. It stores the data permanently unless it is erased. Examples: Floppy disk storage media, Hard disk, CD / DVD, Pen drive, Memory chips etc.,

35. Define number system.

The term computer numbering formats refers to the schemes implemented in digital computer and calculator hardware and software to represent numbers.

36. Define State.

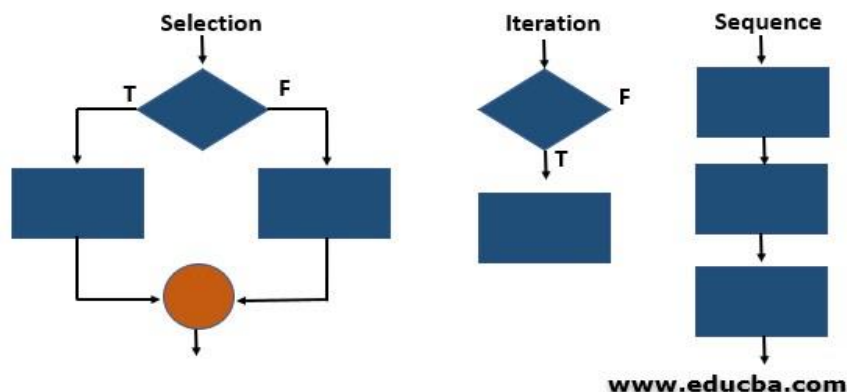
Transition from one process to another process under specified condition with in a time is called state.

37. Define Control Flow.

The process of executing the individual statements in a given order is called control flow.

The control can be executed in three ways

- 1.sequence
- 2.selection
- 3.iteration



38. What is Sequence in algorithm?

All the instructions are executed one after another is called sequence execution.

39. What is Iteration in algorithm?

In some programs, certain set of statements are executed again and again based upon conditional test. i.e. executed more than one time. This type of execution is called looping or iteration.

40. What is Selection in algorithm?

A selection statement causes the program control to be transferred to a specific part of the program based upon the condition.

If the conditional test is true, one part of the program will be executed, otherwise it will execute the other part of the program.

41. Define Function.

Function is a sub program which consists of block of code(set of instructions) that performs a particular task.

For complex problems, the problem is been divided into smaller and simpler tasks during algorithm design.

42. Write the benefits of using Function.

1. Reduction in line of code
2. code reuse
3. Better readability
4. Information hiding
5. Easy to debug and test
6. Improved maintainability

43. Differentiate analog and digital computers? (JAN 2010)

S No	Analog Computer	Digital Computer
1	Process measured data	Process discrete data
2	Analog computers are not precise	Digital computers are more precise
3	Processing speed is low.	Processing speed is high.
4	Less accuracy.	More accuracy.

44. 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.

45. 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.

46. What are the rules for writing pseudo code? (MAY 2010)

- Write on statement per line.
- Capitalize initial keywords.
- Indent to show hierarchy.
- End multi line structure.
- Keep statements language independent.

45. What is recursion?

A function that calls itself is known as recursion. Recursion is a process by which a function calls itself repeatedly until some specified condition has been satisfied.

UNIT II

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

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

1. int
2. float
3. char
4. Double

2. 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.

3. What is an Operator and Operand?

An operator is a symbol that specifies an operation to be performed on operands.

Example: *, +, -, / are called arithmetic operators.

The data items that operators act upon are called operands.

Example: a+b; In this statement a and b are called operands.

4. What is Ternary operators or Conditional operators?

Ternary operators is a conditional operator with symbols ? and :

Syntax: variable = exp1 ? exp2 : exp3

If the exp1 is true variable takes value of exp2. If the exp2 is false, variable takes the value of exp3.

5. What are the Bitwise operators available in 'C'?

& - Bitwise AND

| - Bitwise OR

~ - One's Complement

>> - Right shift

<< - Left shift

^ - Bitwise XOR are called bit field operators

6. What are the logical operators available in 'C'?

The logical operators available in 'C' are

&& - Logical AND

|| - Logical OR

! - Logical NOT

7. What is the difference between Logical AND and Bitwise AND?

Logical AND (&&): Only used in conjunction with two expressions, to test more than one condition. If both the conditions are true the returns 1. If false then return 0. AND (&): Only used in Bitwise manipulation. It is a unary operator.

8. 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.

9. 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.

10. What is conversion specification?

The conversion specifications are used to accept or display the data using the INPUT / OUTPUT statements.

11. What is the difference between 'a' and "a"?

'a' is a character constant and "a" is a string.

12. What is the difference between while loop and do...while loop?

In the while loop the condition is first executed. If the condition is true then it executes the body of the loop. When the condition is false it comes out of the loop. In the do...while loop first the statement is executed and then the condition is checked. The do...while loop will execute at least one time even though the condition is false at the very first time.

13. What is a Modulo Operator?

'%' is modulo operator. It gives the remainder of an integer division

Example: a=17, b=6. Then c=%b gives 5.

14. How many bytes are occupied by the int, char, float, long int and double?

int - 2 Bytes

char - 1 Byte

float - 4 Bytes

long int - 4 Bytes

double - 8 Bytes

15. 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

16. 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:

```
a=5;
```

```
x=a++; /* assign x=5*/
```

```
y=a; /*now y assigns y=6*/
```

```
x=++a; /*assigns x=7*/
```

17. What is a String?

String is an array of characters.

18. What is a global variable?

The global variable is a variable that is declared outside of all the functions. The global variable is stored in memory, the default value is zero. Scope of this variable is available in all the functions. Life as long as the program's execution doesn't come to an end.

19. What are the Escape Sequences present in 'C'?

\n - New Line

\b - Backspace

\t - Form feed

\' - Single quote

\\ - Backspace

\t - Tab

\r - Carriage return

\a - Alert

\” - Double quotes

20. Construct an infinite loop using while?

```
while (1)
```

```
{}
```

Here 1 is a non zero, value so the condition is always true. So it is an infinite loop.

21. Write the limitations of getchar() and scanf() functions for reading strings (JAN 2009)

getchar()

To read a single character from stdin, then getchar() is the appropriate.

scanf()

scanf() allows to read more than just a single character at a time.

22. 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’.

23. What is meant by Control String in Input/Output Statements?

Control Statements contains the format code characters, specifies the type of data that the user accessed within the Input/Output statements.

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

```
main()
```

```
{ {
```

```
float a; float a;
```

```
int x=6, y=4; int x=6, y=4;
```

```
a=x\y; a=(float) x\y;
```

```
printf(“Value of a=%f”, a); printf(“Value of a=%f”,a);
```

```
} }
```

Output 1. 1.500000

25. What is the output of the following program when, the name given with spaces?

```
main()
```

```
{
```

```
char name[50];
```

```
printf(“\n name\n”);
```

```
scanf(“%s, name);
```

```
printf(“%s”,name);
```

```
}
```

Output:

Lachi (It only accepts the data upto the spaces)

26. What is the difference between while(a) and while(!a)?

while(a) means while(a!=0)

while(!a) means while(a==0)

27. Why we don't use the symbol '&' symbol, while reading a String through scanf()?

The '&' is not used in scanf() while reading string, because the character variable itself specifies as a base address.

Example: name, &name[0] both the declarations are same.

28. What is the output of the program?

```
main() increment()
{ {
increment(); static int i=1;
increment(); printf(“%d\n”,i)
increment(); i=i+1;
} }
```

Output:

1 2 3

29. 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.

All statements should be written in lower case letters. Upper case letters are only for symbolic constants. Blank spaces may be inserted between the words. This improves the readability of statements. It is a free-form language; we can write statements anywhere between '{' and '}'. a = b + c; d = b*c;

30. 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 percent_1 class_total.

31. List the types of operators.

- Arithmetic operators
- Relational operators
- Logical operators
- Increment and Decrement operators
- Assignment operators
- Bitwise operators
- Comma operator
- Conditional operator

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

While..DO

- (i) Executes the statements within the while block if only the condition is true.
- (ii) The condition is checked at the beginning of the loop

DO..while

- (i) Executes the statements within the while block at least once.'
- (ii) The condition is checked at the end of the loop.

33. Give the syntax for the 'for' loop statement

```
for (Initialize counter; Test condition; Increment / Decrement)
{
    statements;
}
```

34. What is the use of sizeof() operator?

The sizeof () operator gives the bytes occupied by a variable. No of bytes occupied varies from variable to variable depending upon its data types.

Example:

```
int x,y;
printf(“%d”,sizeof(x));
Output:
2
```

35. What is a loop control statement?

Many tasks done with the help of a computer are repetitive in nature. Such tasks can be done with loop control statements.

36. What are global variable in 'C'?

This section declares some variables that are used in more than one function. such variable are called as global variables. It should be declared outside all functions.

37. Write a program to swap the values of two variables (without temporary variable).

```
#include <stdio.h>
#include <conio.h>
void main( )
{
int a =5; b = 10;
clrscr( );
printf(“Before swapping a = %d b = %d “, a , b);
a = a + b;
B = a – b;
a = a – b;
printf(“After swapping a = %d b = %d”, a,b);
getch( );
}
```

Output:

Before swapping a = 5 b = 10

After swapping a = 10 b = 5

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

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

UNIT - III

1. What is an array?

An array is a group of similar data types stored under a common name.

```
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. What is dynamic memory allocation?

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

4. 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.

5. 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.

6. Where in memory are variables stored?

Variables can be stored in several places in memory, depending on their lifetime.

(1) Variables that are defined outside any function (whether of global or file static scope), and variables that are defined inside a function as static variables, exist for the lifetime of the program's execution. These variables are stored in the data segment. The data segment is a fixed-size area in memory set aside for these variables.

(2) Variables that are the arguments functions exist only during the execution of that function. These variables are stored on the stack. The stack is an area of memory that starts out as small and grows automatically up to some predefined limit.

(3) The third area is the one that does not actually store variables but can be used to store data pointed to by variables. Pointer variables that are assigned to the result of a call to the function malloc() contain the address of a dynamically allocated area of memory. This memory is in an area called the heap.

7. What is an heap memory?

The heap is another area that starts out as small and grows, but it grows only when the programmer explicitly calls malloc() or other memory allocation functions, such as calloc(). The heap can share a memory segment with either the data segment or the stack, or it can have its own segment, it all depends on the compiler options and operating system. The heap, like the stack, has a limit on how much it can grow, and the same rules apply as to how that limit is determined.

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

Array:

1.Array allocates space automatically.

2.It cannot be resized.

PROBLEM SOLVING AND C PROGRAMMING

- 3.It cannot be reassigned.
- 4.Size of(array name) gives the number of bytes occupied by the array.

Pointer:

- 1.Pointer is explicitly assigned to point to an allocated space.
- 2.It can be resized using realloc ().
- 3.Pointers can be reassigned.
- 4.Sezeof(pointer name) returns the number of bytes used to store the pointer variable

9. Comparison of arrays and structures is as follows.

ARRAY:

An array is a collection of data items of same data type.

Arrays can only be declared.

There is no keyword for arrays.

An array name represents the address of the starting element.

An array cannot have bit fields.

STRUTURE:

A structure is a collection of data items of different data types.

Structures can be declared and defined.

The deyword for structures is struct.

A structrure name is known as tag. It is a shorthand notation of the declaration.

A structure may contain bit fields.

10. List the characteristics of Arrays.

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

11. What are the types of Arrays?

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

12. What is the use of '\0' character?

When declaring character arrays (strings), '\0' (NULL) character is automatically added at end. The '\0' character acts as an end of character array.

13. Define Strings.

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

14. Write an example for declaring ARRAYS.

Arrays may consist of any of the valid data types. Arrays are declared along with all other variables in the declaration section of the program.

PROBLEM SOLVING AND C PROGRAMMING

```
/* Introducing array's */
#include <stdio.h>
main()
{
int numbers[100];
float averages[20];
numbers[2] = 10;
--numbers[2];
printf("The 3rd element of array numbers is %d\n", numbers[2]);
}
```

15. Write an example for assigning values to ARRAYS.

The declaration is preceded by the word static. The initial values are enclosed in braces,

Example

```
#include <stdio.h>
main()
{
int x;
static int values[] = { 1,2,3,4,5,6,7,8,9 };
static char word[] = { 'H','e','l','l','o' };
for( x = 0; x < 9; ++x )
printf("Values [%d] is %d\n", x, values[x]);
}
```

16. Define Multi-dimentional array.

Multi-dimensioned arrays have two or more index values which specify the element in the array.

```
multi[i][j];
DECLARATION:
int m1[10][10];
static int m2[2][2] = { {0,1}, {2,3} };
```

17. Example for CHARACTER ARRAYS [STRINGS]

```
#include <stdio.h>
main()
{
static char name1[] = {'H','e','l','l','o'};
static char name2[] = "Hello";
printf("%s\n", name1);
printf("%s\n", name2);
}
```

UNIT IV

1. What is meant by Recursive function?

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

2. 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;
```

3. 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 saves data storage space in memory

4. 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.

5. What are * and & operators means?

'*' operator means 'value at the address'

'&' operator means 'address of'

6. What is meant by Preprocessor?

Preprocessor is the program, that process our source program before the compilation.

7. How can you return more than one value from a function?

A Function returns only one value. By using pointer we can return more than one value.

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

Array name

Type and Size

9. List the header files in 'C' language.

<stdio.h> contains standard I/O functions

<ctype.h> contains character handling functions

<stdlib.h> contains general utility functions

<string.h> contains string manipulation functions

<math.h> contains mathematical functions

<time.h> contains time manipulation functions

10. What are the steps involved in program development life cycle?

1. Program Design
2. Program Coding
3. Program Testing & Debugging

11. What are the types of errors occurred in C program?

1. Syntax errors
2. Runtime errors
3. Logical errors
4. Latent errors

12. What is testing?

Testing is the process of executing the program with sample or tested data.

13. What are the types of testing?

Human testing

Computer based testing

14. How do you define enumerated data type?

```
enum mar_status  
{ single,married,widow };  
enum mar_status person1,person2;  
person1=married;
```

Here the person1 is assigned to value zero.

15. What is meant by debugging?

Debugging is the process of locating and isolating the errors.

16. Specify any five syntax error messages.

Missing semicolon

Missing braces

Missing quotes

Improper comment characters

Undeclared variables

17. What are the pre-processor directives?

Macro Inclusion

Conditional Inclusion

File Inclusion

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.

20. What is the difference between declaring a variable and defining a variable?

Declaring a variable means describing its type to the compiler but not allocating any space for it. Defining a variable means declaring it and also allocating space to hold the variable. A variable can also be initialized at the time it is defined. To put it simply, a declaration says to the compiler, "Some where in the program there will be a variable with this name, and this is the

PROBLEM SOLVING AND C PROGRAMMING

kind of data type it is.” On the other hand, a definition says, “Right here is this variable with this name and this data type”. Note that a variable can be declared any number of times, but it must be defined. Exactly once. For this reason, definitions do not belong in header files, where they might get #included into more than one place in a program.

21. Why does `n++` execute than `n=n+1`?

The expression `n++` requires a single machine instruction such as `INR` to carry out the increment operation whereas; `n+1` requires more instructions to carry out this operation.

22. 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.

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

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

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

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

Some salient points about `main()` are as follows:

1. It is the starting function .
2. It returns an `int` value to the environment that called the program.
3. Recursive call is allowed for `main()` also.
4. It is a user-defined function.
5. Program execution ends when the closing brace of the function `main()` is reached.
6. It has two arguments (a) argument count and (b)argument vector (represents strings passed.)
7. Any user-defined name can also be used as parameters for `main()` instead of `argc` and `argv`

25. 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.

26. 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.

27. Define 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', '\0' };
```

The character of a string are stored in contiguous memory locations as follows:

```
C O L L E G E \0
```

```
1000 1001 1002 1003 1004 1005 1006 1007
```

28. What is the use of 'typedef'?

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

Syntax: typedef data type name;

Example:

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

29. What is 'C' functions? Why they are used?

A function is a self-contained block (or) a sub-program of one or more statements that performs a special task when called. To perform a task repetitively then it is not necessary to re-write the particular block of the program again and again. The function defined can be used for any number of times to perform the task.

30. Differentiate library functions and User-defined functions.

Library Functions	User-defined Functions
a) Library functions are pre-defined set of functions that are defined in C libraries. b) User can only use the function but cannot change (or) modify this function.	a) The User-defined functions are the functions defined by the user according to his/her requirement. b) User can use this type of function. User can also modify this function.

PROBLEM SOLVING AND C PROGRAMMING

Unit - V

1. What is meant by Preprocessor?

Preprocessor is the program, that process our source program before the compilation

2. What are the pre-processor directives?

- Macro Inclusion
- Conditional Inclusion
- File Inclusion

3. Compare arrays and structures.

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

4. Compare structures and unions.

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

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 be handled, functions are more suited, additionally; macros are expanded inline, which means that the code is replicated for each occurrence of a macro.

6. Define Union. Explain Union in detail. (JAN 2009)

Union is a collection of variables similar to structure. The union requires bytes that are equal to number of bytes required for the largest number.

Example:

```
union student
{
char name[20];
int rollno,m1,m2,m3,tot;
float avg;
}s1;
```

Union of structure

Union can be nested with another union.

7. Define Structures. Explain structures in detail. (JAN 2009 / MAY 2009)

PROBLEM SOLVING AND C PROGRAMMING

A structure is a collection of one or more variables of different data types grouped together under a single name. It contains different data types.

Syntax:

```
{
    type variable 1;
    type variable 2;
    type variable n;
} structure_variables;
```

Example:

```
struct student
{
    char name[25];
    int rollno;
    int m1,m2,m3,total;
    float avg;
}s1,s2;
```

- Structure within structure
- Array of structures
- Pointers to structures
- Structures and functions

8. Describe in detail about the Preprocessors in C. (MAY 2009)

The define statement is used to make programs more readable, and allow the inclusion of macros. Consider the following examples,

```
#define TRUE 1 /* Do not use a semi-colon , # must be first character on line */
#define FALSE 0
#define NULL 0
#define AND &
#define OR |#define EQUALS ==
game_over = TRUE;
while( list_pointer != NULL )
```

9. What are pointers? When and why they are used? Explain in detail with sample programs. (JAN 2009/MAY 2009)

Pointer variable is needed to store the memory address of any variable. Denoted by(*) asterisk.

Pointer Declaration:

Syntax:

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

Exmample:

```
int *a;
```

Pointers and Arrays

Pointers and Strings

Pointer as function arguments

Pointer too pointer

Example program:

To add two numbers through variables and their pointers.

To assign a pointer value to another variable.