	E.G.S. PILLAY ENGINEERING COLLEGE (An Autonomous Institution, Affiliated to Anna University, Chennai) Nagore Post, Nagapattinam – 611 002, Tamilnadu.	Rev.0 COE/2017/QB

17CA02102 PROBLEM SOLVING AND PROGRAMMING				
Academic Year :	2018-2019	**Question Bank**	Programme :	P.G-MCA
Year / Semester :	I/I		Course Coordinator:	A.HEMA

Course Objectives	**Course Outcomes:**
1. To understand the various problem solving techniques. 2. To understand the usage of top down design technique in problem solving. 3. To learn the syntax of C. 4. To get exposed to the file processing techniques of C. 5. To get familiarized with the pre-processor directives.	On completion of the course, students will be able to CO1: Design and Implement C programs for a given problem.. CO2: Work with existing programs and modify it as per the Requirements. CO3: Identify the errors in a C program. CO4: Identify the output of a C program without actually executing it. CO5: Develop programs using advanced concepts in c

PART – A (2 Mark Questions With Key)				
S.No	**Questions**	**Mark**	**COs**	**BTL**
UNIT I – PROBLEM SOLVING				
1	**Justify the need for analysis of an algorithm?**		1	K1
	Qualitative and quantitative measure of an algorithm performance, the complexity of algorithms.	1		
	It can be analysis in terms of time efficiency or space utilization.	1		
2	**Define problem solving?**		1	K1
	Problem solving is the process of transforming the Description of a problem in to the solution.	1		
	To select and use appropriate problem –solving strategies, techniques and tools	1		
3	**How can we identify the complexities in any algorithm?**		1	K1
	It can be identified in terms of: space complexity- Memory Usage	1		
	time complexity-Cpu Time	1		
4	**What is an asymptotic notation?**		1	K1
	It used to make meaningful statements about the efficiency of a program.	1		
	It describes the behavior of the time or space complexity for large instance characteristics**.**	1		

5	**Why is an algorithm important in problem solving?**		1	K1
	What our program is going to perform.	1		
	It states some of the actions to be executed.	1		
6	**What is meant by program verification?**		1	K1
	The process of ensuring that a program meets user requirements.	1		
	One of the techniques is the program testing.	1		
7	**List Out the Problem solving strategies.**			
	Divide and conquer ,Dynamic programming	1	1	K1
	Stepwise refinement,Tree,Integer Programming.	1		
8	**Given an Examples of Divide and conquer technique**		1	K1
	Merge sort, Tower of Hanoi	2		
9	**What is top down design?**		1	K1
	It is a Problem-solving technique.	1		
	The problem is divided into sub problems then the process is applied to each sub problem.	1		
10	**Define Stepwise refinement.**		1	K1
	Is a process of breaking down the problem at each stage to obtain a solutions.	2		
11	**What are the factors required for analysis of algorithm?**		1	K1
	Computational Complexity, The order notation, Worst and average case behavior, Probabilistic average case analysis**.**	2		
12	**Write an algorithm for swapping two integers.**		1	K2
	Given two variables a and b, exchange the values assigned to them.	.5		
	Save the original value of a in "t",	.5		
	Assign to "a" the original value to "b"	.5		
	Assign to "b" the original value of "a" that is stored in "t"	.5		
13	**Define an Algorithm.**		1	K1
	An algorithm is a sequence of finite number of steps arranged in a specific logical order, which when executed produces the solution for a problem	2		
14	**Given an Examples of dynamic programming.**		1	K1
	Examples: Knapsack problem, Shortest path ,Multistage graph	2		
15	**List out the types of asymptotic notation**		1	K1
	Big Oh notation Omega notation Theta notation Little Oh notation	2		
UNIT II – BASICS OF C PROGRAMMING				
1	**Define a program.**		1	K1

	The task of processing of data is accomplished by executing sequences of precise.		2		
2	**What are Keywords?**				
	Keywords are certain reserved words that have standard and pre-defined meaning in 'C'.		2	1	K1
3	**List out various types of C tokens.**			1	K1
	Keywords, Constants, Identifiers, Strings, Special symbols and operators**.**		2		
4	**What is meant by type definition?**			1	K1
	Type definition allows user to define an identifier that would represent all existing datatype.		1		
	It takes the form, typedef type identifier;Where type refers to an existing data type and identifier refers to the new name given to the data type.		1		
5	**What are Ternary operators or Conditional operators ?**			1	K1
	Ternary operators is a conditional operator with symbols ? and :		1		
	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.		1		
6	**What is the difference between Logical AND and Bitwise AND?**				
	Logical AND (&&)	**AND (&)**	**2**	1	K1
	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.	Only used in Bitwise manipulation. It is a unary operator			
7	**What is the difference between if and while statement**				
	if	**while**		1	K1
	(i) It is a conditional statement	(i) It is a loop control statement	**.5**		
	(ii) If the condition is true, it executes	(ii) Executes the statements within the	**.5**		
	(iii) If the condition is false then it stops	(iii) If the condition is false the control is	**.5**		

	if(condition)
	{
	True statement;
	}
		while(condition)
		{
		Statement;
		}
			.5		

8	**How many bytes are occupied by the int, char, float, long int and double?**			1	

	int - 2 Bytes,char - 1 Byte,float - 4 Bytes,long int - 4 Bytes,double - 8 Bytes.	**2**		
9	**What is type casting?**		1	K1
	Type casting is the process of converting the value of an expression to a particular datatype.	**1**		
	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.	**1**		
10	**What is the difference between ++a and a++?**		1	K2
	++a means do the increment before the operation (pre increment) a++ means do the increment after the operation (post increment)	**1**		
	Example: a=5; x=a++; /* assign x=5*/ y=a; /*now y assigns y=6*/ x=++a; /*assigns x=7*/	**1**		
11	**Write a c program to check the given number is even or odd.**		1	K2

	#include<stdio.h>
	Main()
	{
	Int I;
	If(i%2==0)
	Printf("the given number is even");
	Else
	Printf("the given number is odd");
	}
		2		

12	**Compare continue and break.**		3	K2
	Continue: It is the loop continuation statement. Whenever this statement is executed, the loop execution will be continued. / Break: It is the loop termination statement. Whenever this statement is executed, the control will come out of the loop.	**2**		
13	**List out the Escape sequences present in 'C'.**		1	K1
	\n – New line , \t – Tab, \b – Back space, \r – Carriage return, \f – Form feed \a – Alert, '- Single quote, \" – Double quotes, \\ - Backspace.	**2**		
14	**Write the result of %.2f and %6.2f for the floating point number.**			
	%.*2f* prints as *floating point*, 2 characters after the point.	**1**	1	K2
	%*6.2f* prints as *floating point*, at least 6 characters wide and 2 characters after the point	**1**		
15	**What is the difference between 'a' and "a"?**		1	K1
	'a' is a character constant and "a" is a string	**2**		
UNIT III – FUNCTIONS,ARRAYS AND STRINGS				
1	**What is meant by a function?**		2	K1

	A function is a subroutine that may include one or more statements Designed to perform a specific task	**1**		
2	**Define main() function.**	**1**		
	Every C program must have one main() function section.	**1**	1	K1
	This section contains two parts, declaration part and executable part.	**1**		
3	**What is meant by recursion?**		1	K1
	Recursion is a process in which a function calls itself.	**1**		

	Eg:
	Main ()
	{
	Printf ("recursion");
	Main ();
	}
		1		

4	**What will happen when you access the array more than its dimension?**		2	K2
	When you access the array more than its dimensions some garbage value is stored in the array.	**1**		
5	**Write the limitations of get char( ) and scanf( ) functions for reading strings**			
	getchar( ): To read a single character from stdin, then getchar() is the appropriate.	**1**	2	K2
	scanf( ): scanf( ) allows to read more than just a single character at a time.	**1**		
6	**What is the difference between scanf() and gets() function?**		2	K2
	In scanf() when there is a blank was typed, the scanf() assumes that it is an end.	**1**		
	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'	**1**		
7	**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);** **} }**		2	K2
	Output:1 .500000	**2**		
8	**Why we don't use the symbol '&' symbol, while reading a String**		2	K2

	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**.**	**2**		
9	**List few limitations of an array.**		1	K2
	• Only same data type is used. • The number of values cannot be altered. • Addition and deletion is very difficult. • It is time consuming as it causes data movement**.**	**2**		
10	**What is function Prototype?**			
	It is a single line description of the function.	**1**	2	K2
	Features: It declares the function It ends with a semicolon The declaration needn't include parameters	**1**		
11	**How can string be represented in c?**			
	A *string* is an array of char objects .string constant, or dynamically allocated memory (see Memory Allocation). store a *null pointer* in the pointer variable.	**1**	2	K2
	The null pointer does not point anywhere, so attempting to reference the string it points to gets an error.	**1**		
12	**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);** **}**		2	K2
	Output: Lachi (It only accepts the data upto the spaces)	**2**		
13	**What is meant by Control String in Input/Output Statements?**		2	K2
	Control Statements contains the format code characters, specifies the type of data that the user accessed within the Input/Output statements.	**2**		
14	**What is the use of a strstr () function?**		2	K2
	It is a two-parameter function that can be used to locate a substring in a string. The general form is, Strstr (s1,s2);	**2**		
15	**Define Return Statement.**		2	K2
	Return statement is used for returning a value from function definition to calling function.	**1**		
	Syntax return (expression);	**1**		

	For example: return a;			
UNIT IV – POINTERS AND AGGREGATE DATA TYPES				
1	**Define Recursion.**			
	Recursion is a repetitive process in which a function call itself. A recursive function is defined recursively whenever the function appears within the definition itself.	2	2	K1
2	**Define delimeters?**		2	K1
	A delimiter is a unique character or series of characters that indicates the beginning or end of a specific statement, string or function body set.	2		
3	**Write the syntax of bitfields?**			

	The declaration of a bit-field has the form inside a structure:
	struct
	{
	type [member_name] : width ;
	};
		2	2	K2

4	**What is pointer declaration?Illustrate.**		1	K1
	Data_type * pt_name; Where,it tells the compiler to do the following, Tells that the variable pt_name is a pointer variable Pt_name needs a memory Pt_name points to a variable of type data_type.	2		
5	**Write a note on enumeration.**		2	K2
	ENUM is closely related to the #define preprocessor. It allows you to define a list of aliases which represent integer numbers.	1		
	enum week { Mon=1, Tue, Wed, Thu, Fri Sat, Sun} days;	1		
6	**What are the ways to access the member of structure variables?**		2	K1
	Using dot notation : V.X Using indirect notation : (*ptr).X Using selection notation : ptr - > X	2		
7	**How to declare a float pointer?**		2	K2

	float *n;
	float m = 9.2;
	n = &m;
	cout << *n << endl;
		1		

	Output: 9.2	1		
8	**What is Pointer indirection?**		1	K1
	The dereference operator or indirection operator, denoted by " * " (i.e. an asterisk), is a unary operator found in C-like languages that include pointer variables**..**	2		
	How does a structure differ from an array?			

9	**structure**	**Array**		2	K2
	Structure can have elements of different data types.	An array is a collection of related data elements of same type.	1		
	A structure is a programmer-defined one.	An array is derived data type.	1		
10	**What is a dangling pointer?**			1	K1
	A dangling pointer is a pointer that has not been initialized. The pointer could be pointing to an unallocated memory or an inaccessible memory.		1		
11	**How one dimensional array is initialized?**			2	K2
	Ex: int a [0]={10,- 50,20,300,5};		1		
	Ex: int a[10]; For (i=0;i<10;i++) Scanf("%d",&a[i]);		1		
12	**How are elements of an array accessed?**			2	K2
	Elements of an array are accessed using subscripts.		2		
13	**What is a subscript? Which is the smallest subscript?**			2	K1
	A subscript or an index is a positive integer value that identifies the storage position of an element in the array.		1		
	0		1		
14	**Define getw and putw functions:**			2	K1
	These are integer-oriented functions. They are similar to get c and putc functions andare used to read and write integer values.		1		
	putw(integer,fp); getw(fp);		1		
15	**What are the methods used to pass the structure functions ?**				
	Passing by value (passing actual value as argument)		1	1	K1
	Passing by reference (passing address of an argument)		1		
UNIT V – FILES AND PREPROCESSOR DIRECTIVES					
1	**What is a file?**			1	K1
	A file is a collection of data that is available in permanent storage.		2		
2	**Write the syntax for file declaration.**			2	K2
	syntax: FILE *filepointer; Ex: FILE *fp;		2		
3	**What are modes in file?**			1	K1
	Mode tells about the types operations like read, write or append that can be performed on a file that is being opened.		2		
4	**Write the syntax to open a file.**			2	K2
	syntax: filepointer=fopen(FILENAME,MODE); Ex: fp=fopen("in.dat",r);		2		
5	**What is the significance of fclose() function?**			1	K1
	This function closes a file that has been opened for an operation. Syntax:		1		
	fclose(filepointer);		1		

	Ex: fclose(fp);			
6	**Write the syntax of fscanf() and fprintf().**		2	K1
	fscanf()- This function is used to read data from a file. Syntax: fscanf(filepointer,"format specifier",&v1,&v2,….);	1		
	fprintf()- this function is used to write data in to a file. Syntax: fprintf(fileprinter,"format specifier",v1,v2,…);	1		
7	**Define : rewind(),fseek()**			
	Rewind() this function is used to reset the FILE pointer to the beginning of the stream regardless of current of the file pointer. The syntax is Void rewind(FILE *fp);	1	1	K1
	Fseek() It is used to move file pointer of the stream to different location.The syntax is Int fseek(FILE * fp,long offset,int origin)	1		
8	**Define Dynamic memory allocation.**		1	K1
	It allows a program to obtain more memory space, while running or to release space when no space is required.	2		
9	**Write the syntax of Malloc().**		2	K1
	ptr=(cast-type*)malloc(byte-size)	2		
10	**Difference between malloc() and calloc()**		2	K2
	malloc() / **calloc()**			
	Allocates requested size of bytes aeturns a pointer first byte of allocated space / Allocates space for an array elements, initializes to zero and then returns a pointer to memory	1		
	ptr=(cast-type*)malloc(byte-size) / ptr=(cast-type*)calloc(n,element-size);	1		
11	**Define Pre-processor.**		1	K1
	It is a program that processor the source code before it passes to the compiler.	2		
12	**What are the pre-processor directives?**		1	K1
	Macro Inclusion. Conditional inclusion. File inclusion**.**	2		
13	**List out the dynamic memory allocation function.**		2	K2
	Malloc(),calloc(),free(),realloc()	2		
14	**Define Conditional inclusion.**		1	K1
	It is used to control the preprocessor with conditional statements**.**	2		
15	**What is command line arguments?**		1	K1
	It allows to pass the information when we ran the program.	2		

PART – B (12 Mark Questions with Key)				
S.No	**Questions**	**Mark**	**COs**	**BTL**

UNIT I – PROBLEM SOLVING				
1	**Discuss the Programs and requirements for problem solving.**	12		
	Programs and Algorithms	3		
	Requirements for problem solving by computer	3	2	K2
	Explanation	3		
	Examples	3		
2	**Explain Problem Solving Aspects in detail.**	12	2	K2
	List out the steps	3		
	Explanation	6		
	Examples	3		
3	**Write about Top Down Design in detail.**	12	1	K1
	Explanation with diagrams	6		
	Breaking a problem into sub p roblems	3		
	Construction Loops	3		
4	**Explain Program Verification in detail.**	12	2	K2
	Explanation	4		
	Steps	4		
	Expressions and equation	4		
5	**Write about The efficiency of algorithms in detail.**	12	1	K1
	Measurement factors	4		
	Steps	2		
	Explanation with examples	6		
6	**Explain The Analysis of algorithms in detail.**	12		
	quantitative measurements	3	2	K2
	Qualitative measurements	3		
	Explanation	3		
	Best,Worst and Average case Analysis	3		
UNIT II - BASICS OF C PROGRAMMING				
1	**Explain about the various datatype in c with an example?**	12		
	Define data type	2		
	List the data type	2	2	K2
	Explanation	6		
	Examples	2		
2	**List out the different operators in c with an example**	12	2	K2
	Define Operators	2		
	Type of operators	2		
	Explanation	4		
	Examples with Program	4		
3	**Write an algorithm, flowchart and C program to check whether a number entered by user is even or odd**	12	2	K2
	Algorithm	4		
	Flowchart	4		

	Program	4		
4	**Describe the control statements of C Program with an example**	12	2	K2
	Name the Control Statements	3		
	Syntax of each control statements	3		
	Explanation	3		
	Example programs	3		
5	**Write a C program to find the factorial of a number.**	12	2	K2
	Problem Solving logic	4		
	Coding	4		
	Explanation with output	4		
6	**Describe the various constant in C.**	12	2	K2
	Define constant	3		
	Types of Constant	3		
	Explanation	6		
UNIT III - FUNCTIONS,ARRAYS AND STRINGS				
1	**Write a C Program to sort an array of elements**	12	2	K2
	Problem Solving Logic	4		
	Coding	4		
	Explanation with Output	4		
2	**Explain the input /output function in c.**	12	2	K2
	List the input function.	3		
	List the output function.	3		
	Explanation with an examples	6		
3	**Write a C program to multiply two matrices**.	12	2	K2
	Logic	4		
	Coding	4		
	Explanation with an examples	4		
4	**Explain about various function prototype**.	12	2	K2
	Define function	2		
	Types of function prototype	2		
	Explanation with an examples	8		
5	**Write short notes on various string handling function with an example.**	12	2	K2
	List the name of the string handling functions.	2		
	Explanation with each one	5		
	Examples of each functions	5		
6	**Explain the various storage classes used in c.**	12	2	K2
	Purpose of storage classes	2		
	Type of storage classes	2		
	Syntax and examples of each one	4		
	Explanation of each one	4		
UNIT IV – POINTERS AND AGGREGATE DATA TYPES				

1	**Develop a structure program called student would contain name,register number and marks of five subjects and percentage.This program to read the details of name,register number and marks of five subjects for 20 students ,Calculate the percentage and display the name,register number,marks of five subjects and percentage of each student.**	12	3	K3
	Problem solving method	4		
	Coding	4		
	Explanation and output	4		
2	**Compare structure and array with an example.**	12	3	K4
	Definition of each one	4		
	Syntax of each one	4		
	Explanation and examples	4		
3	**Comparison between call by value () and call by reference ().**	12	3	K4
	Call by value	4		
	Call by reference	4		
	Explanation of each one	4		
4	**Develop a program to count number of vowels, consonants, digits, spaces and other characters in a line of text using C.**	12	3	K3
	Coding	6		
	Explanation with an examples	6		
5	**Write short notes on pointer with an examples**	12	4	K2
	Pointer definition	2		
	Syntax	2		
	Coding	4		
	Program explanation	4		
6	**Write a program to demonstrate malloc() and calloc() function.**	12	4	K2
	Definition of malloc() and calloc()	4		
	Coding	4		
	Program Explanation	4		
UNIT V – FILES AND PREPROCESSOR DIRECTIVES				
1	**Explain the file handling functions in C( ).**	12	5	K2
	Name of the functions.	2		
	Explanation of each one.	4		
	Program explanation	4		
2	**Develop a program to read a series of integer numbers from the file DATA and the write all odd numbers into the file ODD and even numbers into the file EVEN using C**.	12	5	K3
	Problem Solving method	4		
	program	4		
	Program Explanation	4		
3	**Explain the dynamic memory allocation functions.**	12	5	K2
	Malloc()	3		

	Calloc()	3		
	Free()	3		
	Realloc()	3		
4	**Explain the various file access methods in C**	12	5	K2
	Name the methods	3		
	Working principles	3		
	Explanation	3		
	examples	3		
5	**Explain the preprocessor directives in C with an example**	12	5	K2
	Definition of preprocessor	3		
	Types of preprocessor	3		
	Syntax of each one	3		
	Explanation with an examples	3		
6	**Write short note on sequencial file access functions in C**	12	5	K2
	Logic	6		
	Explanation with an examples	6		

PART – C (20 Mark Questions with Key)				
S.No	**Questions**	**Mark**	**COs**	**BTL**
UNIT I – PROBLEM SOLVING				
1	**Analyze the efficiency of algorithm**	20		
	Analyzing the factors	5		
	Explanation	5	2	K4
	Measurements of efficiency	5		
	Analyze with an examples	5		
2	**Develop a program for the given 2 positive non zero integers n and m.Find GCD of n and m using C.**	20	3	K3
	Algorithm development	8		
	Program	6		
	Explanation	6		
UNIT II - BASICS OF C PROGRAMMING				
1	**(i)Compare switch case and if statement with an examples.**	10		
	(ii)Compare break and Continue statement with an examples.	10		
	(i)Syntax of switch and if	5	2	K4
	Explanation with an examples	5		
	(ii)syntax break and continue	5		
	Explanation with an examples	5		
2	**(i)Develop a program to find the roots of the quadratic equations using C**	10	2	K3
	(ii) Develop a program to print the reverse of a given number	10		
	(i)Problem solving method	5		

	Program Explanation	5		
	(ii)problem solving logic	5		
	Program Explanation	5		
UNIT III - FUNCTIONS,ARRAYS AND STRINGS				
1	**(i)Write a Program to check, whether the given number is prime or not.**	10	3	K2
	(ii) Write a program to accept 5 numbers and print whether the number is even or odd.	10		
	(i)program	5		
	Program explanation and output	5		
	(ii) program	5		
	Program explanation with output	5		
2	**Explain the significance of Array Techniques with an examples**	**20**	**3**	**K2**
	Definition of array	**5**		
	Types of array	**5**		
	Syntax and logic	**5**		
	Explanation	**5**		
UNIT IV – POINTERS AND AGGREGATE DATA TYPES				
1	**(i)Write a program using pointers to read in an array of integers and print its elements in reverse order.**	10	4	K2
	(ii) write short notes on Structures within structures	10		
	(i)program	5		
	Program Explanation	5		
	(ii) structure	5		
	Syntax of nested syntax	5		
2	**(i)Compare structure and union**	10	4	K4
	(ii) compare recursion and iteration	10		
	Definition of structure and union	5		
	Syntax and exmples	5		
	definitions	5		
	Syntax and examples	5		
UNIT V – FILES AND PREPROCESSOR DIRECTIVES				
1	**(i)How can Accessing structure member through pointer using dynamic memory allocation.**	10	4	K2
	(ii)write short notes on typedef and enumerated type.	10		
	Methods	3		
	Declaration and explanation	7		
	Definitions	3		
	Declaration and explanation	7		
2	**Explain the file management in c**	20	5	K2
	File operations	5		
	File mode	5		

	Access methods	5		
	Explanation, examples,syntax	5		