|  |  |
| --- | --- |
| **DHANALAKSHMI SRINIVASAN COLLEGE OF ENGINEERING AND TECHNOLOGY** |  |
| **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING** | |

**dscet-logo**

**EC8393 FUNDAMENTALS OF DATASTRUCTURES IN C**

**UNIT-1**

**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 iscommon 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
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.

**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.

**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

1. **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:**

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

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

**Statement**

if statement

if...else statement

nested if statements

switch statement

nested switch statements

**Description**

An **if statement** consists of a boolean expression followed by one or more statements.

An **if statement** can be followed by an optional **else statement**, which executes when the boolean expression is false.

You can use one **if** or **else if** statement inside another **if** or **else** **if** statement(s).

A **switch** statement allows a variable to be tested for equality against a list of values.

You can use one **swicth** statement inside another **switch** statement(s).

**15.What do you meant by conditional or ternary operator?**

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

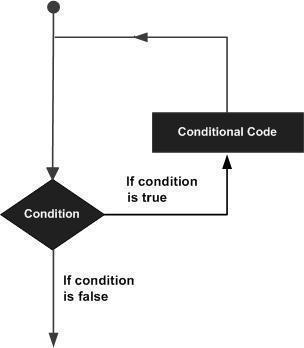
**16. What is the use of sizeof() operator in C.**

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 from of a loop statement in most of the programming languages:



**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 



1. **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

**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);

}

main()

{

float a;

int x=6, y=4;

a=(float) x\y;

printf(“Value of a=%f”,a);

}

**Output:**

**Output:**

1. 1.500000

**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 \\

**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 | (ii) Executes the statements within the |
| some statements. | while block if the condition is true. |
|  |  |
| (iii) If the condition is false then it stops | (iii) If the condition is false the control is |
| the execution the statements. | 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.

**PART – B**

**[First Half]**

**STRUCTURE OF C PROGRAM**

1. Discuss about the structure of C program in detail.[D]

**COMPILATION AND LINKING PROGRAM PROCESS**

2. Explain about the compilation and linking process[D]

**CONSTANTS, VARIABLES AND DATA TYPES**

3. What are Constants? Explain the various types of constants in C.[D]**(April/May 2015)**

**DIFFERENT DATA TYPES IN C**

4. Explain the various data types available in C with example. (Or) Discuss the basic data types in C.**[D](May/June 2016, 2018)**

**OPERATORS AND EXPRESSION**

5.Explain the different types of operators available in C with example.[D]**(April/May 2015) (Nov/Dec 2015) (May/June 2016)**

**MANAGING INPUT / OUTPUT FUNCTIONS IN C**

6.Describe the various input and output statements in C with suitable examples[D]**(April / May 2017)**

**DECISION MAKING**

7.With an example program explain the various decision making statements available in C.[D]**(Nov / Dec 2017)**

8.Explain in detail about various decision making structures available in C with illustrative examples.[D]**(Nov/Dec 2015)**

**BRANCHING AND LOOPING**

9.Describe the various looping statements used in C with suitable examples.[D]**(April/May 2015)**

10.Write about the need and types of branching statements in C language and discuss with examples.**[ID](Dec/Jan 2014)**

**[Second Half]**

**ARRAY**

11.What is an array? Write a C program to arrange the given 10 numbers in ascending order using one dimensional array [D]**(Nov / Dec 2017)**

**STRINGS**

12.Explain the concept of strings in detail.[D]**(April / May 2017, 2018)**

13.Explain the various string handling functions in C.[D]**(April/May 2015)**

**SORTING**

14.Explain about sorting in detail with an example.[D]

**SEARCHING TECHNIQUES**

15.Briefly explain about various searching techniques in detail with an example[D]

**MATRIX OPERATIONS**

16.Write a C program to print the sum of two matrices.**[D](Dec/Jan 2014)**

17.Write a C program to multiply two matrices[D].**(Nov/Dec 2014) (Nov/Dec 2015) (May/June 2016) (Nov / Dec 2016)**

18.Write a C program to subtract two matrices and display the resultant matrix.[D]**(May/June 2014)**

**NIT –II**

**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.

1. **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.

1. **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 | 4.Sezeof(pointer name) returns the number of bytes used |
| number of bytes occupied by the array. | to store the pointer variable. |
|  |
|  |  |

1. **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.

1. **What are the types of Arrays?** 1.One-Dimensional Array
2. Two-Dimensional Array
3. Multi-Dimensional Array
4. **Define Strings.**

**Strings:**

The group of characters, digit and symbols 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.

**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

* 1. **= atoi**(string)

**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.

**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.

1. **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 through31 (binary 11101, 11110, and 11111) as dummy keys.

**PART – B**

**[First Half]**

**FUNCTION**

1. Explain different types of functions with suitable example.[D]

2.Explain the following with suitable examples. (i) Function Declaration.[D].**(Dec/Jan 2014)**

3. Explain different function prototypes.[D]**(Nov/Dec 2014)**

**PARAMETER PASSING METHODS**

4. Explain in detail about call by value and call by reference.[D]**(Apr / May 2018)**

5. Write a C program to swap the content of two variables using pointer.[D]**(Nov / Dec 2017)**

**RECURSION**

6. What is recursion? Explain a recursive function with suitable example.[D] **(Nov/Dec 2015)**

**POINTERS**

7. Discuss in detail about pointer with an example.[D]. **(April / May 2017)**

**POINTERS ARITHMETIC**

8. Explain in detail about pointers arithmetic with suitable example.[D]

9**.** Discuss in detail about Tower of Hanoi with a suitable illustration.[D]

**[Second Half]**

S**TRUCTURE**

10. Explain in detail about the structure with an example.[D]**Apr / May 2018**

**STRUCTURE WITHIN STRUCTURE**

11. Explain Structure within Structure (Structure Assignment / Nested Structure).[D]**Apr / May 2018**

12. Write a C program to create a mark sheet for students using structure.[D]**Nov / Dec 2017**

13. Write a C program to store the employee information using structure and search a particular employee using employee number.[ID].**Nov / Dec 2017**

**UNION**

14. Write short notes on union.[D]

**STORAGE CLASS**

15. Explain the concept of storage classes with suitable example.[D]**(April / May 2017, 2018)**

**PREPROCESSOR DIRECTIVES**

16. Discuss about the preprocessor directives in C.[D] **Nov / Dec 2016**

**UNIT-3**

**1. What is meant by an abstract data type?**

An ADT is an object with a generic description independent of implementation details. This description includes a specification of an components from which the object is made and also behavioral details of obj c s.

**2. Advantages and Disadvantages of arrays? Advantages:**

Data accessing is faster

Array‟s are simple-in terms of understanding point and in terms of programming.

**Disadvantages:**

Array size is fixed

Array elements stored continuously

Insertion and deletion of elements in an array is difficult.

|  |  |  |
| --- | --- | --- |
| **3.** | **What is an array?** |  |
|  | Array may be defined | bstr ctly as a finite ordered set of homogenous |
| elements. Finite means there is | | specific number of elements in the array. |
| **4.** | **What is a linked list?** |  |

Linked list is a kind of series of data structures, which are not necessarily adjacent in memory Each structure contains the element and a pointer to a record containing its successor.

**5. What is singly linked list?**

A singly linked list is a linked list, there exists only one link field in each and every node and all nodes are linked together in some sequential manner and this type of linked list is called singly linked list.

**6. What is a doubly linked list?**

In a simple linked list, there will‟topoint be on the next element, where as in a doubly linked list, there will be two pointers one to point the next

element and the other to point the previous element location.

**7. Define double circularly linked list?**

In a doubly linked list, if the last node or pointer of the list, point to the first element of the list, then it is a circularly linked list.

**8. What is the need for the header?**

Header of the linked list is the first element in the list a d it stor s the number of elements in the list. It points to the first data element of the list

**9. Define Polynomial ADT**

A *polynomial* object is a homogeneous ordered list of pairs <*exponent*, *coefficient*>, where each coefficient is unique.

Operations include returning the degree, extracting the coefficient for a given exponent, addition, multiplication, evaluation for a giv n in ut. 10x4+5x2+1

**10. How to search an element in list.**

Searching can be initiated from first node and it is compared with given element one after the other until the sp cifi k y is found or until the end of the list is encountered.

**11. Define Dqueue?**

Dqueue is also d ta structure where elements can be inserted from both ends and deleted from both ends. To implement a dqueue operations using singly linked list o erations erformed insert front, delete front, insert\_rear, delete rear and display functions.

**12. How to implement stack using singly linked list**

Stack is an Last In First Out (LIFO) data structure. Here , elements are inserted from one end called push operation and the same elements are deleted from the same end called pop operation

So, using singly linked list stack operations are performed in the front or other way ew can perform rear end also.

**13.What are the types of Linear linked list?**

Singly linked lists

Circular singly linked lists

Doubly linked lists

Circular doubly linked lists

**14.What are advantages of Linked lists?** Linked lists are dynamic data structures The size is not fixed

Data can store non-continuous memory blocks

Insertion and deletion of nodes are easier and efficient

Complex applications

**PART B**

**[First Half]**

**STACKS**

1. What is stack? Explain the operations of stack?[D] (Nov/Dec 2015)
2. Describe about the stack data structure and its operations.[D]
3. Explain about the implementation of stack using array[D](Apr/May 2014)
4. List out the applications of stack and explain any two.[ID]

**QUEUES**

1. What is queue? Explain the operations of queue[D]
2. Describe about the queue data structure and its operations.[D](Apr/May 2015)
3. Explain about the implementation of queue using array.[D]

**[Second Half]**

**LINKED LISTS**

1. Write a program for polynomial addition using linked list.[D]
2. Describe about the implementation of queue using linked list.[D](Nov/Dec 2016)

**EVALUATION OF EXPRESSIONS**

1. Write a procedure for evaluation the expression using stack and explain with example.[D](Apr/May 2017).
2. Explain about the implementation of linked list.[D]
3. Define array? How do you represent it?[D]

**UNIT-4**

**1. Write down the algorithm for solving Tow rs of Hanoi roblem?** Push parameters and return address on stack.

If the stopping value has been reach d th n pop the stack to return to previous level else move all except the final disc from starting to intermediate needle. Move final discs from start to estination n dle.

Move remaining discs from interme iate to estination needle.

Return to previous level by popping st ck.

**2. What is a Stack ?**

A stack is a non- rimitive line r data structure and is an ordered collection of homogeneous data elements.The other name of stack is Last-in -First-out list.

One of the most useful concepts and frequently used data structure of variable size for problem solving is the stack.

1. **What are the t o operations of Stack?**

PUSH POP

1. **What is a Queue ?**

A Queue is an ordered collection of items from which items may be deleted at one end called the front of the queue and into which tems may be inserted at the other end called rear of the queue.Queue is called as First –in-First-Out(FIFO).

**5. What is a Priority Queue?**

Priority queue is a data structure in which the intrinsic ordering of the elements does determine the results of its basic operations. Ascending and Descending priority queue are the two types of Priority queue.

1. **What are the different ways to implement list?** Simple array implementation of list

Linked list implementation of list cursor implementation of list

1. **What are the postfix and prefix forms of the expression?** A+B\*(C-D)/(P-R)

Postfix form: ABCD-\*PR-/+

Prefix form: +A/\*B-CD-PR

**8. Explain the usage of stack in recursive algorithm im lementation?**

In recursive algorithms, stack data structur s is used to store the return address when a recursive call is encounter d and also to store the values of all the parameters essential to the current state of the proc dure.

**9. Write down the operations that can be one with queue data structure?**

Queue is a first - in -first out list. The operations that can be done with queue are insert and remove.

**10. What is a circul r queue?**

The queue, which wra s around upon reaching the end of the array is called as circular

queue.

**11.Give few examples for data structures?**

Stacks

Queue

Linked list

Trees

Graphs

1. **List out Applications of queue**

Operating systems often maintain a queue of processes that are ready to execute or that are waiting for a particular event to occur.

Computer systems must often provide a “h processes, two programs, or even two

systems. This holding area is usually called a

“buffer” and is often implemented as a queue.

**13. How do you test for an empty queue?**

To test for an empty queue, we have to check whether READ=HEAD wh re REAR is a pointer pointing to the last node in a queue and HEAD is a point r hat pointer to the dummy header.

In the case of array implementation of queue, the condition to be ch ck d for an empty queue is READ<FRONT.

**14.What are applications of stack?**

Conversion of expression

Evaluation of expression

Parentheses matching

Recursion

1. **Define recursion?**

It is a technique and it can be d fin d as any function that calls itself is called recursion. There are some applications which are suitable for only recursion such as, tower of Hanoi, binary tree traversals etc, can be implementing very easily and efficiently.

**16.What are ty es of Queues?**

Simple queue (ordin ry queue)

Circular queue

Double ended queue

Priority queue

**PART- B**

**[First Half]**

**BINARY TREES**

1. Define binary tree. How do you represent the binary tree.[D]

**BINARY TREE REPRESENTATION AND TRAVERSALS**

2. Explain about the binary tree traversals.[D]

**BINARY SEARCH TREES**

3. Describe about the binary search tree and its operations.[D]

**APPLICATIONS OF TREES**

4.What are the applications of tree.[D]

**[Second Half]**

**SET REPRESENTATIONS**

5. Define set. what are the operation scan be performed on set and explain.[D]

**UNION**

6. Write a c program for find operations usingunion[D]

**GRAPH AND ITS REPRESENTATIONS**

7. How do we represent the graph? Write a short on following:[D]

i. Breadth first traversal

ii. Depth first traversal

8. Explain about the graph traversal with example.[D]

UNIT-5

**1. What is meant by Sorting and searching?**

Sorting and searching are fundamentals operations in computer science. Sorting refers to the operation of arranging data in some given order

Searching refers to the operation of searching the particular record from the existing information

1. **What are the types of sorting available in C?** Insertion sort

Merge Sort

Quick Sort

Radix Sort

Heap Sort

Selection sort

Bubble sort

**3. Define Bubble sort.**

Bubble sort is the one of the easiest sorting method. In this method each data item is compared with its neighbor and if it is an descending sorting , then the bigger number is moved to the top of all

The smaller numbers are slowly mov d to the bottom osition, hence it is also called as the exchange sort.

**3.Mention the various types of s arching t chniques in C** Linear search

Binary search

**5. What is linear se rch?**

In Linear Search the list is se rched 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.

**6. What is binary search?**

Binary search is simpler and faster than linear search. Binary search the array to be searched is divided into two parts, one of which is ignored as it will not contain the required element

One essential condition for the binary search is that the array which is to be searched, should be arranged in order.

**7. Define merge sort?**

Merge sort is based on divide and conquer method. It takes the list to be stored and divide it in half to create two unsorted lists.

The two unsorted lists are then sorted and merge to get a sorted list.

**8. Define insertion sort?**

Successive element in the array to be sorted and inserted into its prop r place with respect to the other already sorted element. We start with s cond element and put it in its correct place, so that the first and second elem ts of the array are in order.

**9. Define selection sort?**

It basically determines the minimum or maximum of the lists and swaps it with the element at the index where its supposed to be.

The process is repeated such that the nth minimum or maximum element is swapped with the element at the n-1th ind x of the list.

**10. What is the basic idea of shell sort?**

Shell sort works by comparing elements that are distant rather than adjacent elements in n rr y or list where a jacent elements are compared.

Shell sort uses n increment sequence. The increment size is reduced after each pass until increment size is 1.

**11.** **What is the ur ose of quick sort and advantage?**

The purpose of the quick sort is to move a data item in the correct direction, just enough for to reach its final place in the array.

Quick sort reduces unnecessary swaps and moves an item to a greater distance, in one move.

**12. Define quick sort?**

The quicksort algorithm is fastest when the median of the array is chosen as the pivot value. That is because the resulting partitions are of very similar size.

Each partition splits itself in two and thus the base case is reached very quickly and it follow the divide and conquer strategy.

**13. Advantage of quick sort?**

Quick sort reduces unnecessary swaps and moves an item to a greater distance, in one move.

**14. Define radix sort?**

Radix sort the elements by processing its individual digits. Radix sort processing the digits either by least significant digit(LSD) method or by most significant digit(MSD) method.

Radix sort is a clever and intuitive little sorting algorithm, radix sort puts the elements in order by comparing the digits of the numbers

1. **List out the different types of hashing functions?** The different types of hashing functions are,

The division method The mind square method The folding method Multiplicative hashing Digit analysis

**16.Define hashing?**

Search from that position for n empty location

Use a second hash function.

Use that array location as the header of a linked list of values that hash to this location

**17. Define hash table?**

All the large collection of data are stored in a hash table. The size of the hash table is usually fixed and it is bigger than the number of elements we are going to store.

The load factor defines the ration of the number of data to be stored to the size of the hash table

**18.What are the types of hashing?**

Static hashing-In static hashing the process is carried out without the usage of an index structure.

Dynamic hashing- It allows dynamic allocation of buckets, i.e. according to the demand of database the buckets can be allocated making this approach more efficient.

1. **Define Rehashing?**

Rehashing is technique also called as double hashing used in hash tabl s o resolve hash collisions, cases when two different values to be s arch d for produce the same hash key.

It is a popular collision-resolution technique in open-addressed hash tables.

**PART B**

**[First Half]**

**LINEAR SEARCH**

1. Write a C program for linear search and explain it.[D]

**BINARY SEARCH**

2. Write a C program for binary search and explain it.[D]

3. What is searching?Explain about the different searching techniques with example.[D]

**[Second Half]**

**BUBBLE SORT**

4. Explain about bubble sort and write a c program.[D]

**INSERTION SORT**

5. Explain about insertion sort and write a c program.[D]

**MERGE SORT**

6. Explain aboutmerge sort and write a c program.[D]

**QUICK SORT**

7. Explain aboutquick sort and write a c program.[D]

**HASH TABLES**

8. Explain about the implementation of hash table[D]

9. Explain about the collision resolution techniques[D]