F. Y. B. Sc. (Computer Science)

Practical Workbook (Laboratory Course – II)

Anekant Education Society's

Tuljaram Chaturchand College,

Baramati – 413102 (Pune)

DEPARTMENT OF COMPUTER SCIENCE

Computer Practical Journal

- CERTIFICATE -

University Seat No. Date:-

This is to certify that Mr./ Miss. / Smt. ______________________________

_____________________ has satisfactorily completed the course in

Practical as prescribed by the University of Pune for the F.Y./ S.Y./

T.Y. B.Sc. (Computer Science-Lab Course I / II / III) in the Year 20 – 20

In-Charge	Internal Examiner	External Examiner	Head
(Practical)			Dept of Computer Science

Computer Science : Paper IV: HTML Programming and Advanced ‘C’ Programming Practicals

INDEX

Sr. No.	Assignment	Date	Signature
	Term – I		
1	Creating simple HTML Pages (use of different tags for changing fonts, foreground and background colors etc.)		
2	HTML programming (use of lists, tables)		
3	HTML programming using frames		
4	HTML programming using hyperlinks		
5	HTML programming (Creation of forms)		
6	HTML programming – Case Study 1		
7	HTML programming – Case Study 2		
8	HTML programming – Case Study 3		
	Term – II		
9	Assignment to demonstrate use of pointers		
10	Assignment to demonstrate concept of strings (string & pointers)		
11	Assignment to demonstrate array of strings		
12	Assignment to demonstrate use of bitwise operators		
13	Assignment to demonstrate structures (using array and functions)		
14	Assignment to demonstrate structures and unions		
15	Assignment to demonstrate command line arguments and preprocessor directives		
16	Assignment to demonstrate file handling (text files)		
17	Assignment to demonstrate file handling (binary files and random access to files)		
18	C Programming – Case Study		

Assignment No. 1 : Some HTML tags required to design simple web pages are given below

Tag	Description	Attributes	Example
<!-- ... -->	Allows one to insert a line of browser-invisible comments in the document		<!— Starting My First web page --!>
<HTML> </HTML>	<HTML> tag tells the browser that this is start of the HTML and </HTML> marks its end.		<HTML> Hello world! </HTML>
<HEAD> </HEAD>	Every html page must have a header. < Head> Tag defines the Head Segment of an html document		
<TITLE> </TITLE>	One of the most important parts of a header is title. Title is the small text that appears in title bar of viewer's browser.		<HEAD> <TITLE> My Web page </TITLE> </HEAD>
<BODY> </BODY>	Every web page needs a body in which one can enter web page content	**background**= designates a file to be displayed as Background **bgcolor**="#(hexadecimal color code)" sets the background Color **text**="#(hexadecimal color code)" sets the color of plain text. Text color default is black.	<BODY BGCOLOR="#00FF00" text=”#FF0000”> Page with Green Color and red Text </BODY> Format of color number is RRGGBB so if we write 00FF00 we mean (red=0, green=255, blue=0)
<BR>	A single tag used to break lines	clear=all|left|right Breaks the text and resumes the next line after the specified margin is clear.	line <BR> is broken
<p>	A single tag used to break text. Breaking text with the <p> tag adds vertical spacing		<p> break the line <p>adding extra space
<B> </B>	To make text appear bold		<B>This Text will appear bold</B>
<U> </U>	To make text appear underlined		<U>This Text will appear underlined</U>
<I> </I>	To make text appear italic		<B><I>This text is both Bold and italic</I>
<CENTER> </CENTER>	Centers enclosed Text		<CENTER> Text is centered </CENTER>
<FONT> </FONT>	To change font which affects the Style (color, typeface, and size) of the enclosed text.	color="#(hexadecimal color code)" sets the color. face=*typeface (or list of typefaces) sets* a typeface for the text (if it is on the user’s machine) size=*value* Sets the size of the	<FONT SIZE="5" FACE="ARIAL" COLOR="#00FF00"> How is this ? </FONT>

1. Create an html page with 7 separate lines in different sizes. State size of each line in its text.

2. Create an html page with 7 separate lines in different colors. State color of each line in its text.

3. Create an html page with all the different text styles (bold, italic and underlined) and its combinations on separate lines. State style of each line in its text.

4. Create an html page containing the polynomial expression as follows

   $a_o + a_1x+ a_2x^2 + a_3 x^3$

5. Create an html page with red background with a message “warning” in large size blinking. Add scrolling text “read the message” below it.

6. Create an html page with following specifications
   a. Title should be about myself
   b. Color the background with pink color
   c. Place your name at the top of the page in large text and centered
   d. Add names of your family members each in a different size, color, style and typeface e. Add scrolling text with a message of your choice
   e. Add your image at the bottom

7. Create an html page with following specifications
   a. Title should be about mycollege
   b. Put the windows Logo image in the background
   c. Place your College name at the top of the page in large text followed by address in smaller size
   d. Add names of courses offered each in a different color, style and typeface
   e. Add scrolling text with a message of your choice
   f. Add college image at the bottom

8. Create an html page with following specifications
   a. Title should be about myCity
   b. Place your City name at the top of the page in large text and in blue color
   c. Add names of landmarks in your city each in a different color, style and typeface
   d. One of the landmark, your college name should be blinking
   e. Add scrolling text with a message of your choice
   f. Add some image at the bottom

_________________________   ___________________   _____________________

Signature of the instructor   Remark   Date

Assignment No. 2 : HTML programming (use of lists, tables)

Tag	Description	Attributes	Example
<LI>	Specify the list item.		
<OL> </OL>	The <OL> tag formats The contents of ar ordered list with numbers. The numbering starts at 1. It is incremented by One for each successive ordered list item tagged with <LI>	Type = a/A/i/I/1 Sets the numbering style to a,A,i, I,1 default 1 start = “A” Specifies the number or letter with which the list should start.	<body bgcolor= "pink"> <font face = "Arial” size= "6" color = "green"> <u> List of Cities.... </u> </font> <ol type = "A" start = "A"> <li> Mumbai <li> Pune <li> Nashik <li> Nagpur </ol> </body>
<UL> </UL>	<UL> tag defines the unordered list of items	Type = disc/square/circle Specifies the bullet type.	<body bgcolor= "sky blue" text =”yellow”> <font face = "Arial” size="6" color= "orange"> <i><u><b> List of Fruits </i></u></b> <ul type = "square"> <li> Apple <li> Pinapple <li> Mango <li> Guava </ul> </body>

Tags used to create table are given in the following table.

Tag	Description	Attributes
<TABLE> </TABLE>	Create a table	Border=number Draws an outline around the table rows and cells of width equal to number. By default table have no borders number =0. Width=number Defines width of the table. Cellspacing=number Sets the amount of cell space between table cells. Default value is 2 Cellpadding=number Sets the amount of cell space, in number of pixels between the cellborder and its contents. Default is 2 Bgcolor=”#rrggbb” sets background color of the table Bordercolor=”#rrggbb” sets border color of the table align=left|right|center Aligns the table. The default alignment is left frame=void|above|below|hsides|lhs|rhs|vsides|box|border Tells the browser where to draw borders around the table
<TR> </TR>	Creates a row in the table	

<TH> </TH>	Cells Are inserted in a row of the table for heading	
<TD> </TD>	Data cells are inserted in a row of the table	

□1 . Write the HTML code which generates the following output. □

- □ Coffee
- □ Tea
  - ▪ Black Tea
  - ▪ Green Tea
    1) Africa
    2) China
- □ Milk

2. Write the HTML code which generates the following output. □

Country	Population (In Crores)	
INDIA	1998	85
	1999	90
	2000	100
USA	1998	30
	1999	35
	2000	40
UK	1998	25
	1999	30
	2000	35

Signature of the instructor　　　　Remark　　　　Date

Assignment No. 3 : HTML programming using frames

Frames : Using frames, one can divide the screen into multiple scrolling sections, each of which can display a different web page into it. It allows multiple HTML documents to be seen concurrently

Tags used to add frames are given in the following table.

Tag	Description	Attributes	Example
<FRAMESET> </FRAMESET>	Splits browser screen into frames.	Rows=number helps in dividing the browser screen into horizontal sections or frames. Cols=number divides the screen into vertical sections or frames. The number written in the rows and cols attribute can be given as absolute numbers or Percentage Value or an asterisk can be used to indicate the remaining space.	<frameset rows = “20%, 30%, *”>
<FRAME> </FRAME>	used to define a single frame in a <frameset>	name=*text* Assigns a name to the frame Noresize Prevents users from resizing the frame. src=*url* Specifies the location of the initial HTML file to be displayed by the frame. bordercolor=*”#rrggbb”* or *color name* Sets the color for frame’s borders	<html> <frameset rows = "50%, *"> <frameset cols = "50%, *"> <frame src = "success.html" name = "frm1"> <frame src = welcome.html"> </frameset> <frame src = "failure.html"> </frameset> </html>

1. Divide the frame into different sections as shown below and add appropriate html files to each frame.

<table>
<tr><td colspan="6">First Frame : Name and Address</td></tr>
<tr><td colspan="3">Second Frame
Bulleted list of qualifications</td><td colspan="3">Third Frame
Links to Favourite sites</td></tr>
<tr><td colspan="2">Fourth Frame
Scrolling Message</td><td colspan="2">Fifth Frame
Blinking reminders</td><td colspan="2">Sixth Frame
image</td></tr>
</table>

2. Create an HTML page with two horizontal frames containing Heading and other information. Add a bulleted list of your favourite subjects in one & in other, for each subject make a nested list that contains teacher name, the start and end time.
3. Create an HTML page with two vertical frames containing Heading and other information. Add an ordered list of your educational qualifications in one & in other, for each course make a nested list that contains university or board name, the year and the percentage scored.

________________________ ____________________ ____________________

Signature of the instructor Remark Date

Assignment No. 4 : HTML programming using hyperlinks

Hyperlinks : Hyperlink is a specialized feature of HTML. Instead of clicking through sequentially organized pages, a hypertext user clicks specially highlighted text called 'hyperlink'. Hyperlinks are technically known as anchors. They are usually visible in blue underlines.

Tags used to add hyperlinks lists are given in the following table.

Tag	Description	Attributes	Example
<A> </A>	Add an anchor or hyperlink.	href=*url* Specifies the URL of the target page.	<BODY> <A HREF="http://www.yahoo.com">Click here to visit Yahoo</A> </BODY>

1. Create an HTML page with two horizontal frames containing Heading and other information. Add a bulleted list of your favourite subjects in one & in other for each subject make a nested list that contains teacher name, the start and end time. Add image and message in a separate frame Add link to teacher or college web site wherever teacher name appears.
2. Create an HTML page with two vertical frames containing Heading and other information. Add an ordered list of your educational qualifications or each course make a nested list that contains university or board name, the year and the percentage scored in one and in other, Add link to university site where university name appears. Add your college photograph and message in a separate frame.

________________________ ____________________ ______________________

Signature of the instructor Remark Date

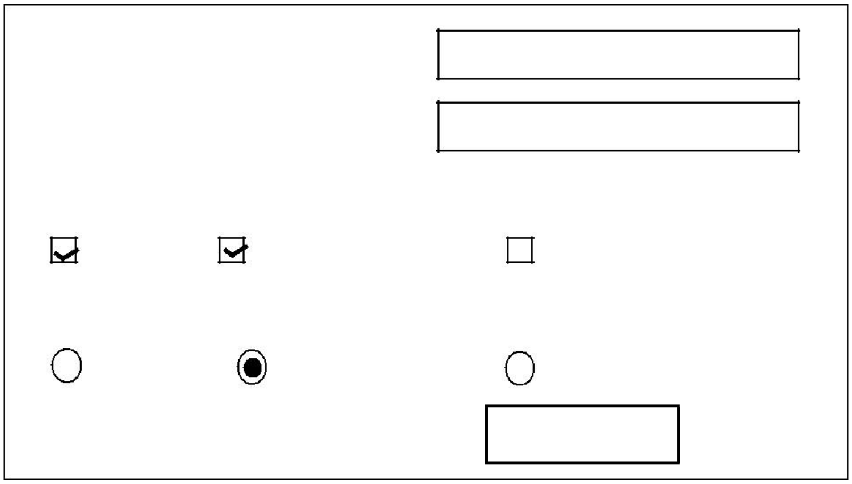
Assignment No. 5 : HTML programming (Creation of forms)

Tags used to add input forms are given in the following table.

Tag	Description	Attributes	Example
<FORM> </FORM>	Creates a form	action="URL" Gives the URL of the application that is to receive & process the forms data. method="get" or "post" Setsthe method by which the browser sends the forms data to the server for processing.	<html> <body bgcolor = "pink"> <form name = "sample"> Enter The name &nbsp;:<br> <br>Enter Salary &nbsp; &nbsp &nbsp;&nbsp;: </body> </html>
<INPUT> </INPUT>	It is used for managing the input controls that will be placed within the tag.	Name="text" It is used to name the field. Maxlength=number The maximum number of input characters allowed in the input control. Size=number The width of the input control in pixels. type="(checkbox/ hidden/ radio /reset /submit /text /image)" value="default" value to be submitted with the form (for a checkbox or radio button); or label (for Reset or Submit buttons)" src="source file for an image", checked Indicates that checkbox or radio button is checked	<body bgcolor = "pink"> <form name = "sample"> Enter The name : <input type = "text" name = "nm" width = 30><br> <br>Enter Salary &nbsp; &nbsp; &nbsp; : <input type = "text" name = "sal" width = 10><br> <br>Gender &nbsp; &nbsp; &nbsp;&nbsp;&nbsp; &nbsp;&nbsp; &nbsp;:: <input type = "radio" name = "gender" value = "male" checked> Male <input type = "radio" name ="gender" value = "female" > Female<br> </body>
<SELECT> </SELECT>	Defines and displays a set of optional list items from which the user can select one	Name="(name to be passed to the script as part of name/value pair)" rows="no. of rows" cols="(no. of cols)"	 <br>Age between : <select name="age" size=1></select><br>

	or more items.		
<OPTION>	Indicates a possible item within a select widget	Selected=default selection value=”data submitted if this option is selected”	<select name=”age” size=1> <option selected>21-30 <option>31-40 <option>41-50 <option>51-60 <option>61-70 </select>
<TEXTAREA> </TEXTAREA>	Multiline text entry widget	Name= name of data field size=# of items to display. Multiple allows multiple selections	<br> Objectives &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <textarea rows=10 columns=15></textarea>

1 . Design the following form in HTML.



Enter Your Name

Enter Your Password

Which of the following Operating System have you used?

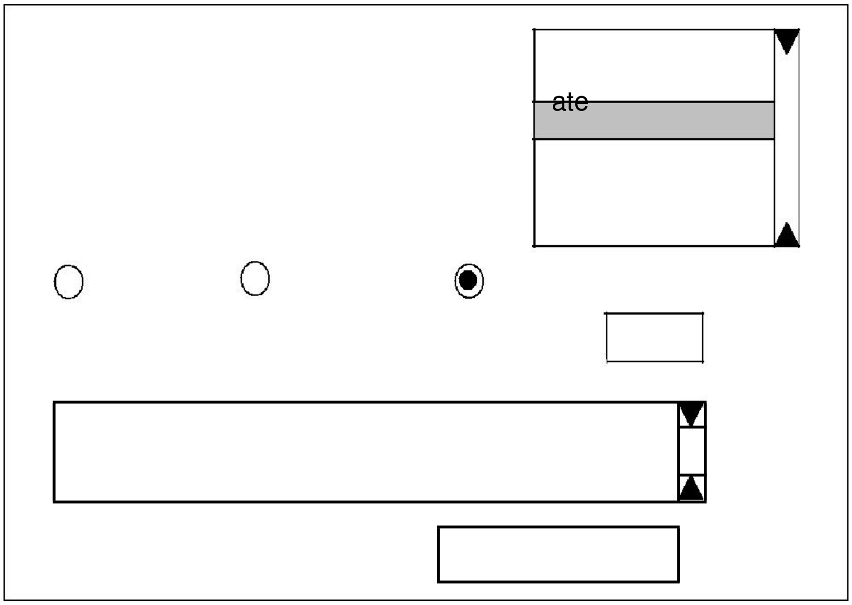
LINUX Windows
XP
Macintos
h 8.0

Which Operating System do you like the best?

LINUX Windows XP Macintosh 8.0

You have Completed the Form . **Submit**

2. Design the following form in HTML.

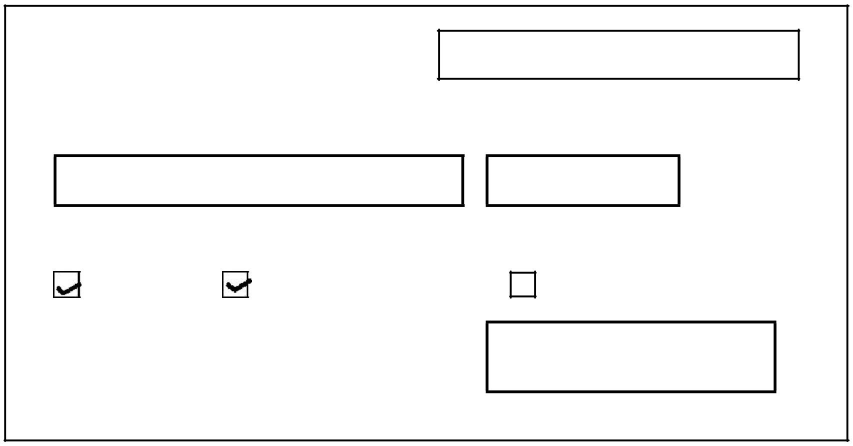


Vanilla

Chocal Pistachio

Mango

Santra Mantra



Choose your favourite ice cream flavour

How would you like to have it?

CUP CONE BAR

How Many people would you like to serve?

Tell Us something about your self

To clear the contents click. **Reset**

3. Design the following form in HTML.

Enter Name of your friend

Choose the file you want to post to your friend

What does the file contain?

Browse…

Image Source code Binary code

You have Completed the Form . **Submit Query**

Signature of the instructor Remark Date

Assignment No. 6 : HTML programming – Case Study 1

Design a website for college which contains homepage showing college's basic details. The homepage should contain the overall sitemap of the college including hyperlinks to the various streams(Arts, Commerce, Science) and the departments which are existing under that stream. It should also display the information of staff members serving in their respective departments.

________________________ ___________________ _____________________

Signature of the instructor Remark Date

Assignment No. 7 : HTML programming – Case Study 2

Design a website for College Library which contains homepage showing Library's basic information. The homepage should contain the overall sitemap of the library including hyperlinks to the various sections which are existing. It should also display the various books according to their section type(novel, biography, autobiography, short story, poetry etc.). Also display the information of new arrivals in library using <marquee> tag.

________________________   ___________________   _____________________

Signature of the instructor   Remark   Date

Assignment No. 8 : HTML programming – Case Study 3

Design a website for a supermaket which contains homepage showing supermaket's basic details. The homepage should contain the overall sitemap of the supermaket including hyperlinks to the various departments which are existing. It should also display the various items according to their category(clothing, stationary, electronic, home appliances etc.). Display proper images.

_______________________ _________________ ___________________

Signature of the instructor Remark Date

Assignment No. 9 : Assignment to demonstrate use of pointers

A Pointer is a variable that stores the memory address of another variable

Actions involving Pointers	**syntax**	**Example**

Declaration of pointers	data_type * pointer_name	int *p1,*p2;
		float *temp1;

Initialization of pointers	pointer =&variable
	p1=&n;
		int a, *p= &a;

Pointer Arithmetic	The C language allow arithmetic operations to be performed on pointers: Increment, Decrement, Addition, Subtraction When a pointer is incremented (or decremented) by 1, it increments by sizeof(datatype). For example, an integer pointer increments by sizeof(int).	
Pointers and Functions	We can pass the address of a variable to a function. The function can accept this address in a pointer and use the pointer to access the variable's value.	
Arrays And Pointers	An array name is a pointer to the first element in the array. It holds the base address of the array. Every array expression is converted to pointer expression as follows: a[i] is same as *(a+i)	int n; *n , *(n + 0) represents 0th element n[j], *(n+ j),* (j + n) , j [n] ; represent the value of the jth element of array n

Pointer To Pointer		int a;
		int * p;
		int **q;
		p = &a;
		q = *p ;

To allocate memory Dynamically	The functions used are : malloc, calloc, realloc
	ptr = (cast-type *) malloc (byte-size) ;
	Allocates a block of contiguous bytes. If the space in heap is not sufficient to satisfy request, allocation fails, returns NULL.
	ptr1 = (cast-type *) calloc (byte-size);
	Similar to malloc, but initializes the memory block allocated to 0.
	ptr = realloc (ptr, new size);
	To increase / decrease memory size.
		int * p,*p1;
		p = (int *) malloc(10 * sizeof(int));
		p1 = (int *) calloc(10, sizeof(int));
		p1=realloc(p1,20*sizeof(int));

1.Write a C program to Accept data & display its addresses.

2. Write a C program to Accept array elements & perform sum of array elements using pointers

3.Write a program to allocate memory dynamically for n integer Accept & displays the values.

4. Write a function which takes hours, minutes and seconds as parameters and an integer s and increments the time by s seconds. Accept time and seconds in main and Display the new time in main using the above function.

5. Write a program to display the elements of an array containing n integers in the reverse order using a pointer to the array.

________________________ ________________ ________________

Signature of the instructor Remark Date

Assignment No. 10 : Assignment to demonstrate concept of strings (string & pointers)

A string is an array of characters terminated by a special character called NULL character(\0). Each character is stored in 1 byte as its ASCII code.

Actions Involving strings	**Explanation**	**Example**
Declaring Strings		char message[80];
Initializing Strings		char message[]= { ’H’, ’e’, ’l’, ’l’, ’o’, ’\0’ } ; char message [ ] = “Hello”;

Accepting Strings	scanf and gets can be used to accept strings	char name[20], address[50];
		printf(“\n Enter your name :);
		scanf(“%s”, name);
		printf(“\n Enter your address :);
		gets(address);

Displaying Strings	printf and puts can be used to display strings.	printf(“\n The name is %s:”, name);
		printf(“\n The address is :”);
		puts(address);

String functions	All string operations are performed using functions in “string.h”. Some of the most commonly used functions are
	a. strlen – Returns the number of characters in the string (excluding \0)
	b. strcpy – Copies one string to another
	c. strcmp – Compares two strings. Returns 0 (equal), +ve (first string > second), -ve (first string < second). It is case sensitive
	d. strcmpi – Same as strcmp but ignores case
	e. strcat – Concatenates the second string to the first. Returns the concatenated string.
	f. strrev – Reverses a string and returns the reversed string.
	g. strupr – Converts a string to uppercase.
	h. strlwr – Converts a string to lowercase
		#include <string.h>
		main( )
		{
		char str1[50], str2[50],str3[100];
		printf(“\n Give the first string:”);
		gets(str1);
		printf(“\n Give the second string string:”);
		gets(str2);
		if (strlen(str1) == strlen(str2)
		{strcpy(str3, strrev(str1));
		 strcat(str3, strupr(str2));
		 puts(strupr(str3));
		}
		Else
		   puts(strlwr(str2);
		}

1. Write a menu driven program to perform the following operations on strings using standard library functions:
   Length Copy Concatenation Compare
   Reverse Uppercase Lowercase Check case

2. Write a C program to Accept names from array & search name from array.

3. A palindrome is a string that reads the same-forward and reverse. *Example*: “madam” is a Palindrome. Write a function which accepts a string and returns 1 if the string is a palindrome and 0 otherwise. Use this function in main.

4. For the following standard functions, write corresponding user defined functions and write a menu driven program to use them. strcat, strcmp, strrev, strupr

5. Write a menu driven program which performs the following operations on strings. Write a separate function for each option. Use pointers
   i. Check if one string is a substring of another.
   ii. Count number of occurrences of a character in the string.
   iii. Replace all occurrences of a character by another.

________________________ ____________________ ______________________

Signature of the instructor Remark Date

Assignment No. 11 : Assignment to demonstrate array of strings

An array of strings is a two dimensional array of characters. It can be treated as a 1-D array such that each array element is a string.

Actions Involving array of strings	**Explanation**	**Example**
Declaring String array	char array[size1][size2];	char cities[4][10]
Initializing String array		char cities[4][10] = { “Pune”, “Mumbai”, “Delhi”, “Chennai”};

1. Write a program that accepts n words and outputs them in dictionary order.

2. Write a program that accepts n strings and displays the longest string.

3. Write a program that accepts a sentence and splits the sentence into words. Sort each word and reconstruct the sentence.
Input – this is a string Output – hist is a ginrst

4. Write a function, which displays a given number in words. For Example: 129 One Hundred Twenty Nine
2019 Two Thousand Nineteen

5. Define two constant arrays of strings, one containing country names (ex: India, France etc) and the other containing their capitals. (ex: Delhi, Paris etc). Note that country names and capital names have a one-one correspondence. Accept a country name from the user and display its capital. Example: Input: India , Output: Delhi.

______________________	________________	__________________
Signature of the instructor	Remark	Date

Assignment No. 12 : Assignment to demonstrate use of bitwise operators

Bitwise operators: C provides 6 operators to perform operations on bits. These operators operate on integer and character but not the float and double. Ones complement operator (~) is unary while the others are binary.

Operator	**Purpose**	**Example**
~	One’s complement	~a : Complements each bit of variable a
⍰⍰	Right shift	a=a>>1; Shifts bits of a one position to the right
⍰⍰	Left Shift	a=a<<n; Shifts bits of a n positions to the left
&	Bitwise AND	a = b&c; performs bitwise AND on b and c a = a&0xFF00; Masks the lower order 8 bits of a
|	Bitwise OR	a = a!b; performs bitwise OR on b and c
^	Bitwise XOR	x = x^y; y=x^y; x=x^y; Swaps x and y by performing bitwise XOR.

Sample code: The following function accepts an integer argument and displays it in binary format. It uses shift operator and AND masking.

```
void displaybits(unsigned int n)
{
   unsigned int mask = 32768; /*set MSB of mask to 1
  */ while (mask>0)
  {
     if((n&mask)==0)
          printf(“0”);
    else printf(“1”);
    mask = mask >>1; /* shift mask right */
}
```

1. Write a program to swap two variables without using a temporary variable. (Hint: Use XOR)

2. Write a program which accepts two integers x and y and performs x<<y and x>>y. Display the result in binary.

Signature of the instructor　　　　　　Remark　　　　　　Date

Assignment No. 13 : Assignment to demonstrate structures (using array and functions)

A structure is a composition of variables possibly of different data types, grouped together under a single name. Each variable within the structure is called a 'member'.

Operations performed	**Syntax / Description**	**Example**

Declaring a structure	struct structure-name
	{
	type member-1 ;
	type member-2;
	.
	.
	type member-n ;
	};
		struct student
		{
		char name[20];
		int rollno;
		int marks;
		};

Creating structure variables	struct structurename variable;	struct student stud1;
Accessing structure members	variable.member	stud1.name stud1.rollno stud1.marks
initializing a structure variable	the initialization values have to be given in {} and in order	struct student stud1 = {"ABCD",10,95};

Pointer to a structure	struct structure-name * pointer-name;	struct student *ptr;
		ptr = &stud1;

Accessing members using Pointer	pointer-name -> member-name;	ptr->name; ptr->rollno;
Array of structures	struct structure-name array-name[size];	struct student stud[10];
passing Structures to Functions	return-type function-name (struct structure-name variable);	void display(struct student s);
pass an array of structures to a function	return-type function-name (struct structure-name array[size]);	void display(struct student stud[10]);

1. Write a C program to Accept details for 'n' Student & display information of each Student.

2. Write a C program to accept details of n students(roll no.,name,marks of 3 subjects,percentage) and display student data having more than 70% marks.

3. Write a C program to Accept details for 'n' books & write a menu driven program for
   1)Display all text books
   2)Find total cost of all books

4. Create a structure student (roll number, name, marks of 3 subjects, percentage). Accept details of n students and write a menu driven program to perform the following operations. Write separate functions for the different optiii) Modify
   i) Display all student details
   ii) Display all student having percentage > _____
   iii) Display student having maximum percentage

5. Create a structure employee (id, name, salary). Accept details of n employees and write a menu driven program to perform the following operations. Write separate functions for the different options.

   i) Search by name
   ii) Search by id
   iii) Display all
   iv) Display all employees having salary > ____
   v) Display employee having maximum salary Instructor should fill in the blanks with appropriate values.

Signature of the instructor

Remark

Date

Assignment No. 14 : Assignment to demonstrate structures and unions

The individual members of a structure can be other structures as well. This is called nesting of structures.

Operations performed	**Syntax**	**Example**

Nested Structure	struct structure1
	{
	. . .
	struct structure2
	{
	. . .
	} variable;
	. . .
	};
	
	Method 2
	struct structure2
	{
	. . .
	};
	
	struct structure1
	{
	. . .
	struct structure2 variable;
	. . .
	};
		struct student
		{
		int rollno; char name[20];
		struct date
		{
		int dd, mm, yy;
		} bdate, admdate;
		};
		
		struct date
		{
		int dd, mm, yy;
		};
		
		struct student
		{
		int rollno; char name[20];
		struct date bdate, admdate;
		};

Accessing nested structure members	nested structure members can be accessed using the (.) operator repeatedly.	stud1.bdate.dd, stud1.bdate.mm

Self Referential structure	A structure containing a pointer to the same structure	struct node
		{
		int info;
		struct node *next;
		};

Unions	A union is a variable that contains multiple members of possibly different data types grouped together under a single name. However, only one of the members can be used at a time. They occupy the same memory area.	union u
		{
		char a;
		int b;
		};

```
/* Program for demonstrating structure and union */
struct library_book
{
    int id;
   char title[80],publisher[20] ; int code;
    union u
    {
              int no_of_copies; char month[10];
               int edition;
    }info;
    int cost;
};
void main( )
```

```
        {
        struct library_book book1;
        printf("\n Enter the details of the book \n");

        printf("\n Enter the id, title and publisher \n"); scanf("%d%s%s",&book1.id,
        book1.title, book1.publis her); printf("\n Enter the code: 1-Text Book, 2-
        Magazine, 3-Reference");
           scanf("%d",book1.code);
        switch(book1.code)
        {
           case 1: printf("Enter the number of copies :");
                                   scanf("%d",&book1.info.no_of_copies); break;
       case 2: printf("Enter the issue month name :"); scanf("%s",book1.info.month);
                                   break;
       case 3: printf("Enter the edition number:"); scanf("%d",&book1.info.edition);
                                   break;
   }
        printf("Enter the cost :"); scanf("%d",&book1.cost);
   /* Display details of book */ printf("\n id = %d", book1.id);
   printf("\n Title = %s", book1.title);
   printf("\n Publisher = %s", book1.publisher); switch(book1.code)
   {
     case 1: printf("Copies = %d:", book1.info.no_of_copies); break;
      case 2: printf("Issue month name = %s",book1. info.month); break;
      case 3: printf("Edition number =%d:",book1.info.edition); break;
     }
     printf("\n Cost = %d", book1.cost);
    }
```

1. Modify the demonstration program above to accept details for n books and write a menu driven program for the following:
   i)Display all text books
   ii)Search Text Book according to Title
   iii) Find the total cost of all books (Hint: Use no_of_copies).

2. Modify the demonstration program to accept details for n books and write a menu driven program for the following:

   i)Display all reference books
   ii) Find the total number of reference books
   iii) Display the edition of a specific reference book.

3. Modify the sample program 1 to accept details for n books and write a menu driven program for the following:

   i) Display all magazines
   ii) Display magazine details for specific month.
   iii) Find the "costliest" magazine.

________________________ ____________________ ______________________

Signature of the instructor Remark Date

Assignment No. 15 : Assignment to demonstrate command line arguments and preprocessor directives

Preprocessor directives	They begin with a # which must be the first non-space character on the line. They do not end with a semicolon.	
Macro Substitution Directive	# define MACRO value	# define PI 3.142
Argumented Macro	# define MACRO(argument) Value	# define SQR(x) x*x #define LARGER(x,y) ((x)>(y)?(x):(y))
Nested macro	one macro using another	#define CUBE(x) (SQUARE(x)*(x))

File Inclusion directive	#include <filename>
	#include “filename”
		#include <stdio.h>

Conditional Compilation directive	# if, # else, # elif, # endif #ifdef	#ifdef PI #undef PI #endif

Command Line Arguments	int argc - argument counter
	char *argv[]-argument vector
		void main(int argc, char *argv[])
		{
		printf(“There are %d arguments in all”, argc);
		for (i=0; i<argc; i++)
		printf(”Argument %d =%s”,i,argv[i]);
		}

To run a program using command line arguments	Compile the program using cc Execute the program using a.out followed by command line Arguments	Example: a.out ABC 20 Here, ABC and 20 are the two command line arguments which are stored in the form of strings. To use 20 as an integer, use function atoi . Example: int num = atoi(argv[2]);

1. Write a C program to Display contents of file using command line argument.

2. Write a C program to Copy the contents of one file to another Using command line argument.

3. Write a program to accept two filenames as command line arguments. Copy the contents of the first file to the second such that the case of all alphabets is reversed

4. Define a macro EQUALSTR which compares two strings x and y and gives 1 if equal and 0 otherwise. Use this macro to accept two strings from the user and check if they are equal.

______________________ ______________________ ______________________

Signature of the instructor Remark Date

Assignment No. 16 : Assignment to demonstrate file handling (text files)

Operations performed	**Syntax**	**Example**
Declaring File pointer	FILE * pointer;	FILE *fp;
Opening a File	fopen(“filename”,mode); where mode = “r”, “w”, “a”, “r+”, “w+”, “a+”	fp=fopen(“a.txt”, “r”);

Checking for successful open	if (pointer==NULL)	if(fp==NULL)
		exit(0);

Checking for end of file	Feof	if(feof(fp))
		printf(“File has ended”);

Closing a File	fclose(pointer);
	fcloseall();
		fclose(fp);

Character I/O	fgetc, fscanf
	fputc, fprintf
		ch=fgetc(fp);
		fscanf(fp, ”%c”,&ch);
		fputc(fp,ch);

String I/O	fgets, fscanf
	fputs, fprintf
		fgets(fp,str,80);
		fscanf(fp, ”%s”,str);

Reading and writing formatted data	fscanf
	fprintf
		fscanf(fp, ”%d%s”,&num,str);
		fprintf(fp, “%d\t%s\n”, num, str);

Random access to files	ftell, fseek, rewind	fseek(fp,0,SEEK_END); /* end of file*/ long int size = ftell(fp);

1. Write a C program to Display contents of file using file handling.

2. Write a C program read the contents of file and Count no of characters,blanks,tabs and lines from file.

3. Write a program to accept a filename as command line argument and count the number of words,lines
and characters in the file.

4. Write a program to accept details of n students (roll number, name, percentage) and write it to a file named “student.txt”. Accept roll number from the user and search the student in the file. Also display the student details having the highest percentage.

Signature of the instructor

Remark

Date

Assignment No. 17 : Assignment to demonstrate file handling (binary files and random access to files)

In binary files, information is written in the form of binary . All data is written and read with no interpretation and separation i.e. there are no special characters to mark end of line and end of file.I/O operations on binary files

Reading from a binary file	fread(address,size-of-element,number of elements,pointer);	fread (&num,sizeof(int),1,fp);
		Fread (&emp,sizeof(emp),1,fp);
		fread(arr,sizeof(int),10,fp);

Writing to a binary File	fwrite(address,size-of-element,number of elements,pointer);	fwrite (&num,sizeof(int),1,fp);
		Fwrite (&emp,sizeof(emp),1,fp);

```
/* Program to demonstrate binary file */
struct employee
{ char name[20]; float sal;
}; main( )
{
            FILE *fp;
            struct employee e; int i;
            if((fp=fopen ("employee.in","wb"))==NULL)
                        { printf("Error opening file"); exit( );
                        }

            for(i=0;i<5;i++)
            {
                        printf("\n Enter the name and salary");
                        scanf("%s%f",e.name,&e.sal); fwrite(&e,sizeof(e),1,fp);
            }
          fclose(fp);

          fp=fopen("employee.in","rb");      /*         reopen file */
          if(fp==NULL)
          {         fprintf(stderr, "Error opening file);
          exit( );
          }
          for(i=0;i<5;i++)
          {
                    fread(&e,sizeof(e),1,fp);
                    printf("\n Name = %s Salary = %f",e.name,e.sal);
          }fclose(fp);
}
```

1. Create a structure student (roll number, name, percentage) Write a menu driven program to perform the following operations on a binary file- "student.dat". Write separate functions for the different options.
   1. Add a student (Note: Students should be assigned roll numbers consecutively)
   2. Search Student
      a) according to name
      b) according to roll number
   3. Display all students.

2. Create a structure student (roll number, name, percentage). Write a menu driven program to perform the following operations on a binary file- "student.dat". Write separate functions for the different options.
   a. Add a student (Note: Students will be assigned roll numbers consecutively)
   b. Delete student
      i) according to name
      ii) according to roll number
   c. Display all students.

Signature of the instructor Remark Date

Assignment No. 18 : C Programming – Case Study
(Handling various string operations without built in functions)

Design a program in C to cover the following options:

1. Find the length of the string using pointer.
2. Find whether the given string is present in large string or not
3. Compare two strings without using library functions
4. Count total number of capital and small letters from accepted line using point
5. Concatenate two strings without using library function.

_________________________ _________________________ _________________________

Signature of the instructor Remark Date