Revision-2015 Course Title- Programming in C		Course Code-4044		
Qu. No:	Scoring Indicators	Split up Score	Sub Total	Total
I -1	Identifiers- names given to entities such as variables, functions, structures etc	2	2	10

I -2	do
	{
	//Statements
	
	}while(condition test);
		2	2	

I -3	**pointer** :-variable whose value is the address of another variable, i.e., direct address of the memory location	2	2	
I -4	Functionname(argument1,argument2..)	2	2	
I -5	**function** :- group of statements that together perform a task	2	2	
II-1	Following is the basic structure of a C program. **Documentation**-Consists of comments, some description of the program, programmer name and any other useful points that can be referenced later. **Link**-Provides instruction to the compiler to link function from the library function. **Global declaration**-Consists of function declaration and global variables. **main( )** { }Every C program must have amain() function which is the starting point of the program execution. **Subprograms** **User defined functions.** *Any example program	3 3	6	

II-2	#include<stdio.h>
	voidmain()
	{
	int a, b,**c**;
	printf("**Enter two numbers** to add\n");
	scanf("%d%d", &a, &b);
	c= a + b;
	printf("**Sum** of the **numbers** = %d\n",**c**);
	}
		6	6	

II-3	**Entry Controlled Loop** If Test condition is false, **loop** body will be executed once. Eg:**for loop** and while **loop** **Exit Controlled Loop** is used when checking of test condition is mandatory after executing **the loop** body.eg:do while **loop** *Any example	2 2 2	6	

II-4	#include <stdio.h>
	int main()
	{
	int array[100],search,c,n;
	printf("Enter number of elements in array**n**");
	scanf("%d",&n);
	printf("Enter %d integer(s)**n**",n);
	
	for(c=0;c<n;c++)
	scanf("%d",&array[c]);
	
	printf("Enter a number to search**n**");
	scanf("%d",&search);
	
	for(c=0;c<n;c++)
	{
	if(array[c]==search)/* *If required element is found* */
	{
	printf("%d is present at location %d.**n**",search,c+1);
	break;
	}
	}
	if(c==n)
	printf("%d isn't present in the array.**n**",search);
	return0;
	}
		6	6	

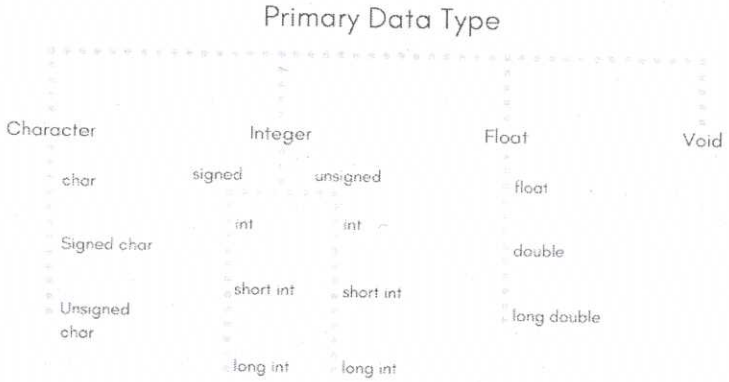
```
Enter the number of elements in array
7
Enter 7 numbers
1
2
4
2
2
5
8
Enter the number to search
2
2 is present at location 2.
2 is present at location 4.
2 is present at location 5.
2 is present 3 times in array.
```

II-5	#include<stdio.h>
	int main()
	{
	int num = 10;
	printf("Value of variable num is: %d", num);
	/* To print the address of a variable we use %p
	* format specifier and ampersand (&) sign just
	* before the variable name like &num.
	*/
	printf("\nAddress of variable num is: %p", &num);
	return0;
	}
	
	Output:
	Value of variable num is: 10
	Address of variable num is: 0x7fff5694dc58
		6	6	

II-6	➢strcat - concatenate two **strings**. ➢strchr **string** scanning operation. ➢strcmp - compare two **strings**. ➢strcpy - copy a **string**. ➢strlen – get **string** length.	6	6	

II-7	#include<stdio.h>
	int addNumbers(int a, int b);
	int main()
	{
	int n1,n2,sum;
	printf("Enters two numbers: ");
	scanf("%d %d",&n1,&n2);
	sum = addNumbers(n1, n2); // function call
	printf("sum = %d",sum);
	return0;
	}
	int addNumbers(int a,int b) // function definition
	{
	int result;
	result = a+b;
	return result; // return statement
	}
		6	6	42

III-(a)		5		



Derived data types: array,stucture,union and pointer.

3

III(b)

```
switch(monthno)
{
        case 1:
            printf("January\n");
            break;
        case 2:
            printf("February\n");
            break;
        case 3:
            printf("March\n");
            break;
        case 4:
            printf("April\n");
            break;
        case 5:
            printf("May\n");
            break;
        case 6:
            printf("June\n");
            break;
        case 7:
```

7

	printf("July\n");
	break;
	case 8:
	printf("August\n");
	break;
	case 9:
	printf("September\n");
	break;
	case 10:
	printf("October\n");
	break;
	case 11:
	printf("November\n");
	break;
	case 12:
	printf("December\n");
	break;
	default:
	printf("invalid Month number. \nPlease try again ....\n");
	break;
	}
	}
			15	30

IV(a)	The two-way selection:- based on resolving an expression, and then executing a set of commands depending on whether the response was true or false. eg:if...else multi-way selection structure :-executes one of many sets of statements, depending on the value of the control expression.eg:- switch statement *example program	5 3		

IV(b)	printf("Enter positive numbers (0 or -ve number to stop):\n");
	sum = 0;
	scanf ("%d", &num);
	while (num > 0) {
	sum += num;
	scanf ("%d", &num);
		7	15	30

V(a)	**Declaration of two dimensional Array in C** data_type array_name[rows][columns]; eg:int twodimen[4][3]; **Initialization of 2D Array in** **Cint** arr[4][3]={{1,2,3},{2,3,4},{3,4,5},{4,5,6}} *example progam	5 3		

V(b)	printf("Enter the location where you wish to insert an element\n");
	scanf("%d",&position);
	
	printf("Enter the value to insert\n");
	scanf("%d",&value);
	
	for(c=n-1;c>=position-1;c--)
	array[c+1]=array[c];
	array[position-1]=value;
	printf("Resultant array is\n");
	for(c=0;c<=n;c++)
	printf("%d\n",array[c]);
		7	15	

VI(a)	•for loop •while loop •do-While loop •break statement •continue statement •goto statement • *example programs	5 3		

VI(b)	printf("Enter the location where you wish to delete element\n");
	scanf("%d",&position);
	if(position>=n+1)
	printf("Deletion not possible.\n");
	else
	{
	for(c=position-1;c<n-1;c++)
	array[c]=array[c+1];
	printf("Resultant array:\n");
	for(c=0;c<n-1;c++)
	printf("%d\n",array[c]);
		7	15	30

VII (a)	**String declaration**:char string_variable_name [array_size];
	eg:char first_name[15] = "ANTHONY";
	
	String Intialization:char first_name[ ] = "NATHAN";
	
	Reading and displaying a string:
	char name[15];
	gets(name); //reads a string
	puts(name); //displays a string
		5		

VII(b)	***example program**
	
	printf("Enter the first string\n");
	gets(a);
	printf("Enter the second string\n");
	gets(b);
	y=strcat(a,b);
	x=strlen(y)
	printf("String obtained on concatenation: %s\n",a);
		3
		
		7
			15	

VIII(a)	There are four arithmetic operators that can be used on pointers: ++, --, +, and - *example programs:-one for increment and one for decrement operation	4		
VIII(b)		4		

	charstring1[100], string2[100];
	scanf("%[^\n]\ns", string1);
	printf("Enter the first string: %s", string1);
	scanf("%[^\n]\ns", string2);
	printf("\nEnter the second string: %s", string2);
	printf("\nAre both strings same: ");
	if(strcmp(string1, string2) == 0) {
	printf("Yes");
	}
	else{
	printf("No");
	}
		7	15	30

IX (a)	**Function Declaration and calling**
	
	int main (){
	
	/* local variable definition */
	int a =100;
	int b =200;
	int ret;
	
	/* calling a function to get max value */
	ret = max(a, b);
	
	printf("Max value is : %d\n", ret);
	
	return0;
	}
	
	/* function returning the max between two numbers */
	int max(int num1,int num2){
	
	/* local variable declaration */
	int result;
	
	if(num1 > num2)
	result = num1;
	else
	result = num2;
	
	return result;
	}
		8		

IX(b)	int data[5], i;
	printf("Enter elements: ");
	for(i = 0; i <5; ++i)
	scanf("%d", data + i);
	printf("You entered: \n");
		7		

	for(i = 0; i <5; ++i) printf("%d\n", *(data + i));		15	

X(a)	**Recursion:**calling a function within itself
	Factorial of a given number
	unsignedlonglongint factorial(unsignedint i)
	{
	if(i <=1){
	return1;
	}
	return i * factorial(i –1);
	}
	int main()
	{
	int i =i2;
	printf("Factorial of %d is %d\n", i, factorial(i));
		8		

X(b)	*#include<stdio.h>*
	void disp(char ch)
	{
	printf("%c ", ch);
	}
	int main()
	{
	char arr[] = {'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j'};
	for (int x=0; x<10; x++)
	{
	
	disp (arr[x]);
	}
	
	return0;
	}
		7	15	30