1A

|  |
| --- |
| Write a program in C to show the simple structure of a function. *Expected Output* :  The total is : 11  Write a program in C to find the square of any number using the function.  Test Data : Input any number for square : 20 Expected Output :  The square of 20 is : 400.00  Write a program in C to swap two numbers using function.   Test Data : Input 1st number : 2 Input 2nd number : 4 Expected Output :  Before swapping: n1 = 2, n2 = 4  After swapping: n1 = 4, n2 = 2 |

2A

|  |
| --- |
| Write a program in C to check whether a number is a prime number or not using the function.   Test Data : Input a positive number : 5 Expected Output :  The number 5 is a prime number.  Write a program in C to get the largest element of an array using the function.   Test Data : Input the number of elements to be stored in the array :5 Input 5 elements in the array : element - 0 : 1 element - 1 : 2 element - 2 : 3 element - 3 : 4 element - 4 : 5 Expected Output :  The largest element in the array is : 5 |

2B

|  |
| --- |
| Write a program in C to check armstrong and perfect numbers using the function  Test Data : Input any number: 371 Expected Output :  The 371 is an Armstrong number.  The 371 is not a Perfect number.  Write a program in C to print all perfect numbers in given range using the function.   Test Data : Input lowest search limit of perfect numbers : 1 Input lowest search limit of perfect numbers : 100 Expected Output :  The perfect numbers between 1 to 100 are :  6 28 |

1B

|  |
| --- |
| Write a program in C to check a given number is even or odd using the function.  Test Data : Input any number : 5 Expected Output :  The entered number is odd.  Write a program in C to find the sum of the series 1!/1+2!/2+3!/3+4!/4+5!/5 using the function.  Expected Output :  The sum of the series is : 34  Write a program in C to convert decimal number to binary number using the function.  Test Data : Input any decimal number : 65 Expected Output :  The Binary value is : 1000001 |

3A

|  |
| --- |
| Write a program in C to check whether two given strings are an anagram.  Test Data : Input the first String : spare Input the second String : pears Expected Output :  spare and pears are Anagram.  Write a C programming to find out maximum and minimum of some values using function which will return an array.Test Data : Input 5 values 25 11 35 65 20 Expected Output :  Number of values you want to input: Input 5 values  Minimum value is: 11  Maximum value is: 65 |

4A

|  |
| --- |
| Write a program in C to convert a tm object to custom wide string textual representation.  Expected Output :  The textual representation of specified date and time :  Sunday 09/02/16 17:51:10  Sunday 09/02/16 17:51:10  Write a program in C to convert a time\_t object to calendar time expressed as Coordinated Universal Time.  Expected Output :  The calendar time expressed as Coordinated Universal Time is :  UTC: Thu Aug 03 10:53:03 2017  local: Thu Aug 03 16:23:03 2017  Write a program in C to convert a time\_t object to calendar time expressed as local time.  Expected Output :  The calendar time expressed as a local Time is :  UTC: Thu Aug 03 11:15:59 2017  local: Thu Aug 03 16:45:59 2017 |

4B

|  |
| --- |
| Write a program in C to print the date and time before 24 months.  Expected Output :  Today is : Thu Aug 3 17:27:16 2017  (DST is not in effect)    24 months ago the date was : Mon Aug 3 17:27:16 2015  (DST was not in effect)  Write a program in C to show the first of calendar time.  Expected Output :  Sun Jan 01 00:00:00 1900  Write a program in C to show the start of the epoch.  Note : epoch means the beginning of a period in the history of someone.  Expected Output :  0 seconds since the epoch began  Thu Jan 01 00:00:00 1970 |

3B

|  |
| --- |
| Write a program in C to print the current time.  Expected Output :  The Current time is : Thu Aug 03 13:38:58 2017  Write a program in C to compute the number of seconds passed since the beginning of the month.  Expected Output :  222084 seconds passed since the beginning of the month.  Write a program in C to convert a time\_t object to a textual representation.  Expected Output :  Thu Aug 03 13:44:49 2017  Write a program in C to convert a tm object to custom textual representation.  Expected Output :  The textual representation of specified date and time :  September Sun Sep 2 16:30:32 2016 pm  September Sun Sep 2 16:30:32 2016 pm |

5A

|  |
| --- |
| ***Output of the program:***  #include<stdio.h>  // function prototype, also called function declaration  void swap(int \*a, int \*b);    int main()  {      int m = 22, n = 44;      //  calling swap function by reference      printf("values before swap m = %d \n and n = %d",m,n);      swap(&m, &n);  }    void swap(int \*a, int \*b)  {      int tmp;      tmp = \*a;      \*a = \*b;      \*b = tmp;      printf("\n values after swap a = %d \nand b = %d", \*a, \*b);  } |

6A

|  |
| --- |
| int main()  {  int x = 2, y = 4;  printf("before swapping x = %d and y = %d\n", x, y);  swap(&x, &y); // call by reference  return 0;  } //end of program |

6B

|  |
| --- |
|  |

5B

|  |
| --- |
| 1. ***Output of the program:***   // arguments pass by reference  #include <stdio.h>  void swap (int \*a, int \*b) // a and b are reference variables  {  int temp;  temp = \*a;  \*a = \*b;  \*b = temp;  } |

7A

|  |
| --- |
|  |

8A

|  |
| --- |
|  |

8B

|  |
| --- |
|  |

7B

|  |
| --- |
|  |

9A

|  |
| --- |
|  |

10A

|  |
| --- |
|  |

10B

|  |
| --- |
|  |

9B

|  |
| --- |
|  |

11A

|  |
| --- |
|  |

12A

|  |
| --- |
|  |

12B

|  |
| --- |
|  |

11B

|  |
| --- |
|  |

13A

|  |
| --- |
|  |

14A

|  |
| --- |
|  |

14B

|  |
| --- |
|  |

13B

|  |
| --- |
|  |

15A

|  |
| --- |
|  |

16A

|  |
| --- |
|  |

16B

|  |
| --- |
|  |

15B

|  |
| --- |
|  |

17A

|  |
| --- |
|  |

18A

|  |
| --- |
|  |

18B

|  |
| --- |
|  |

17B

|  |
| --- |
|  |

19A

|  |
| --- |
|  |

20A

|  |
| --- |
|  |

20B

|  |
| --- |
|  |

19B

|  |
| --- |
|  |

21A

|  |
| --- |
|  |

22A

|  |
| --- |
|  |

22B

|  |
| --- |
|  |

21B

|  |
| --- |
|  |

23A

|  |
| --- |
|  |

24A

|  |
| --- |
|  |

24B

|  |
| --- |
|  |

23B

|  |
| --- |
|  |

25A

|  |
| --- |
|  |

26A

|  |
| --- |
|  |

26B

|  |
| --- |
|  |

25B

|  |
| --- |
|  |

27A

|  |
| --- |
|  |

28A

|  |
| --- |
|  |

28B

|  |
| --- |
|  |

27B

|  |
| --- |
|  |

29A

|  |
| --- |
|  |

30A

|  |
| --- |
|  |

30B

|  |
| --- |
|  |

29B

|  |
| --- |
|  |