PRACTICAL – TEST

PROGRAM -1

AIM- Write a C program to display the n terms of odd natural numbers and their sum.

Test Data

Input number of terms : 10

Expected Output : The odd numbers are :1 3 5 7 9 11 13 15 17 19

The Sum of odd Natural Number upto 10 terms : 100

CODE-

|  |
| --- |
| // Write a C program to display the n terms of odd natural numbers and  // their sum.  // Test Data  // Input number of terms : 10  // Expected Output :  // The odd numbers are :1 3 5 7 9 11 13 15 17 19  // The Sum of odd Natural Number upto 10 terms : 100  #include <stdio.h>  int main()  {  int i, n, sum = 0;  printf("Enter the number of terms: ");  scanf("%d", &n);  printf("The odd numbers are: ");  for (i = 1; i <= n; i++)  {  printf("%d ", 2 \* i - 1);  sum += 2 \* i - 1;  }  printf("\nThe Sum of odd Natural Number upto %d terms : %d\n", n, sum);  } |

OUTPUT-

|  |
| --- |
|  |

PROGRAM -3

AIM- Write a program in C to find largest and smallest number in array

CODE-

|  |
| --- |
| // Write a program in C to find largest and smallest number in array.  #include<stdio.h>  int main()  {  int arr[10],i,n,large,small;  printf("Enter the size of array = ");  scanf("%d",&n);  printf("Enter the elements of array = ");  for(i=0;i<n;i++)  {  scanf("%d",&arr[i]);  }  large=small=arr[0];  for(i=1;i<n;i++)  {  if(arr[i]>large)  {  large=arr[i];  }  if(arr[i]<small)  {  small=arr[i];  }  }  printf("Largest number = %d\n",large);  printf("Smallest number = %d\n",small);  } |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -4

AIM- C Program to delete element from array at given index.

CODE-

|  |
| --- |
| // C Program to delete element from array at given index.  #include<stdio.h>  int main()  {  printf("\n HARSH D \n");  int i,n,a[100],pos;  printf("Enter the size of array = ");  scanf("%d",&n);  printf("Enter the elements of array = ");  for (i=0;i<n;i++)  {  scanf("%d",&a[i]);  }  printf("Enter the position of element to be deleted = ");  scanf("%d",&pos);  if (pos>=n+1)  {  printf("Deletion not possible");  }  else  {  for (i=pos-1;i<n-1;i++)  {  a[i]=a[i+1];  }  printf("The resultant array is = ");  for (i=0;i<n-1;i++)  {  printf("%d ",a[i]);  }  }  return 0;} |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -6

AIM- Write a C a program to calculate sum of elements of an array using pointer.

CODE-

|  |
| --- |
| // Write a C a program to calculate sum of elements of an array using pointer  #include<stdio.h>  int main()  {  printf("\n HARSH D \n");  int array[50],i,n,sum=0;  printf("The Number Of elements in the array = ");  scanf("%d",&n);  printf("Enter the elements of the array = ");  for(i=0;i<n;i++)  {  scanf("%d",&array[i]);  }  for(i=0;i<n;i++)  {  sum=sum+array[i];  }  printf("The Sum of the elements of the array is = %d",sum);  return 0;  } |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -7

AIM- Write a program in C to check a String is palindrome or not.

CODE-

|  |
| --- |
| // Write a program in C to check a String is palindrome or not.  #include <stdio.h>  #include <string.h>  int main()  {  printf("\n HARSH D \n");  char str[100];  int i, len, flag = 0;  printf("Enter a string:");  scanf("%s", str);  len = strlen(str);  for(i=0;i<len/2;i++)  {  if(str[i]!=str[len-i-1])  {  flag=1;  }  }  if(flag==1){  printf("Not a palindrome");  }  else  {  printf("Palindrome");  }  return 0;  } |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -9

AIM- Write a C a program to print the address of character and the character of string using pointer

CODE-

|  |
| --- |
| // Write a C a program to print the address of character and the character of string using pointer  #include<stdio.h>  #include<string.h>  int main()  {  printf("\n HARSH D \n");  char str[100];  char \*ptr;  int i;  printf("Enter a string = ");  scanf("%s",str);  ptr=str;  for(i=0;i<strlen(str);i++)  {  printf("Address of %c is %u\n",\*(ptr+i),ptr+i);  }  return 0;  } |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -10

AIM- Write a C a program for sorting using pointer.

CODE-

|  |
| --- |
| // Write a C a program for sorting using pointer.  #include <stdio.h>  void swap(int \*a, int \*b)  {  int temp = \*a;  \*a = \*b;  \*b = temp;  }  void bubbleSort(int \*arr, int n)  {  int i, j;  for (i = 0; i < n - 1; i++)  {  for (j = 0; j < n - i - 1; j++)  {  if (\*(arr + j) > \*(arr + j + 1))  {  swap((arr + j), (arr + j + 1));  }  }  }  }  int main()  {  printf("\n HARSH D \n");  int n;  printf("Enter the number of elements: ");  scanf("%d", &n);  int arr[n];  printf("Enter the elements: ");  for (int i = 0; i < n; i++)  {  scanf("%d", &arr[i]);  }  bubbleSort(arr, n);  printf("Sorted array: ");  for (int i = 0; i < n; i++)  {  printf("%d ", arr[i]);  }  printf("\n");  return 0;  } |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -11

AIM- Write a program in C to print Floyd's Triangle.

CODE-

|  |
| --- |
| INCOMPLETE |

PROGRAM -2

AIM- Write a program in C to remove duplicate elements form array in C

CODE-

|  |
| --- |
| // Write a program in C to remove duplicate elements form array in C  #include<stdio.h>  int main()  {  printf("\n HARSH D \n");  int arr[100],i,j,n,temp;  printf("Enter the size of array = ");  scanf("%d",&n);  printf("Enter the elements of array = ");  for(i=0;i<n;i++)  {  scanf("%d",&arr[i]);  }  for(i=0;i<n;i++)  {  for(j=i+1;j<n;j++)  {  if(arr[i]==arr[j])  {  for(int k=j;k<n;k++)  {  arr[k]=arr[k+1];  }  n--;  }  }  }  printf("Array after removing duplicate elements = ");  for(i=0;i<n;i++)  {  printf("%d ",arr[i]);  }  return 0;  } |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -5

AIM- Define a structure cricket that will describe the following information: Player name

Team name

Batting average

Using cricket, declare an array player with 50 elements and WAP to read the information about all the 50 players and print team wise list containing names of players with their batting average.

CODE-

|  |
| --- |
| // Define a structure cricket that will describe the following information: Player name  // Team name  // Batting average  // Using cricket, declare an array player with 50 elements and WAP to read the information about all the 50 players and print team wise list containing names of players with their batting average.  #include <stdio.h>  #include <string.h>  struct cricket  {  char name[20];  char team[20];  float batting\_avg;  };  int main()  {  printf("\n HARSH D \n");  struct cricket player[50];  int i,n;  printf("Enter the number of players = ");  scanf("%d",&n);  for(i=0;i<n;i++)  {  printf("Enter the name of the player = ");  scanf("%s",player[i].name);  }  for(i=0;i<n;i++)  {  printf("Enter the name of the team = ");  scanf("%s",player[i].team);  }  for(i=0;i<n;i++)  {  printf("Enter the batting average = ");  scanf("%f",&player[i].batting\_avg);  }  for(i=0;i<n;i++)  {  printf("Name: %s\n",player[i].name);  printf("Team: %s\n",player[i].team);  printf("Batting average: %f\n",player[i].batting\_avg);  }  return 0;  } |

OUTPUT:-

|  |
| --- |
|  |

PROGRAM -2

AIM- Write a program in C to find the smallest missing element in a sorted array.

Expected Output :

The given array is : 0 1 3 4 5 6 7 9 The missing smallest element is: 2

CODE-

|  |
| --- |
| // Write a program in C to find the smallest missing element in a sorted array.  // Expected Output :  // The given array is : 0 1 3 4 5 6 7 9  // The missing smallest element is: 2  #include <stdio.h>  int main()  {  printf("\n HARSH D \n");  int arr[100], n, i, j, temp;  printf("Enter the size of the array: ");  scanf("%d", &n);  printf("Enter the elements of the array: ");  for (i = 0; i < n; i++)  {  scanf("%d", &arr[i]);  }  for (i = 0; i < n; i++)  {  for (j = i + 1; j < n; j++)  {  if (arr[i] > arr[j])  {  temp = arr[i];  arr[i] = arr[j];  arr[j] = temp;  }  }  }  printf("The sorted array is: ");  for (i = 0; i < n; i++)  {  printf("%d ", arr[i]);  }  printf("\n");  for (i = 0; i < n - 1; i++)  {  if (arr[i] + 1 != arr[i + 1])  {  printf("\n The missing smallest element is: %d", arr[i] + 1);  }  }  return 0;  } |

OUTPUT:-

|  |
| --- |
|  |