**PSG COLLEGE OF TECHNOLOGY, COIMBATORE -641004**

**DEPARTMENT OF COMPUTER APPLICATIONS**

**I SEMESTER MCA**

**23MX16 C PROGRAMMING LABORATORY**

**PROBLEM SHEET 3 – ARRAYS – SIMPLE PROBLEMS**

1. Write a program to accept the integer values and display the second largest value in an array.

CODE:

#include <stdio.h>

int main() {

int n,temp;

printf("Enter no of elements:");

scanf("%d",&n);

int arr[n];

for(int i=0;i<n;i++){

printf("Enter the elements in a array:");

scanf("%d",&arr[i]);

}

for(int i=0;i<n;i++){

for(int j=i+1;j<n;j++){

if(arr[i]>arr[j]){

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

printf("Ascending Order:");

for(int i=0;i<n;i++){

printf("%d\t",arr[i]);

}

printf("\nSecond largest %d",arr[1]);

return 0;

}

1. Write a program to sort the list of numbers in an ascending and descending order.

CODE:

#include <stdio.h>

int main() {

int n,temp;

printf("Enter no of elements:");

scanf("%d",&n);

int arr[n];

for(int i=0;i<n;i++){

printf("Enter the elements in a array:");

scanf("%d",&arr[i]);

}

for(int i=0;i<n;i++){

for(int j=i+1;j<n;j++){

if(arr[i]>arr[j]){

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

printf("Ascending Order:");

for(int i=0;i<n;i++){

printf("%d\t",arr[i]);

}

for(int i=0;i<n;i++){

for(int j=i+1;j<n;j++){

if(arr[i]<arr[j]){

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

printf("\nDescending Order:");

for(int i=0;i<n;i++){

printf("%d\t",arr[i]);

}

return 0;

}

1. Write a program to search for a specified number in an array and display with its position.

CODE:

#include <stdio.h>

int main() {

int n,key;

int found=0;

printf("Enter no of elements:");

scanf("%d",&n);

int arr[n];

for(int i=0;i<n;i++){

printf("Enter the elements in a array:");

scanf("%d",&arr[i]);

}

printf("Enter a key element:");

scanf("%d",&key);

for(int i=0;i<n;i++){

if(arr[i]==key){

printf("Element found at position:%d",i);

found=1;

return 0;

}

}

if(found==0){

printf("Element not found");

}

return 0;

}

1. Write a program to find the occurrence of positive, negative, even and odd elements for a given array.

CODE:

#include <stdio.h>

int main() {

int n;

int positive=0,negative=0,odd=0,even=0;

printf("Enter no of elements:");

scanf("%d",&n);

int arr[n];

for(int i=0;i<n;i++){

printf("Enter the elements in a array:");

scanf("%d",&arr[i]);

}

for(int i=0;i<n;i++){

if(arr[i]>0){

positive++;

}

else if(arr[i]<0){

negative++;

}

if(arr[i]%2==0){

even++;

}

else{

odd++;

}

}

printf("Positive numbers count:%d\n",positive);

printf("Negative numbers count:%d\n",negative);

printf("Odd numbers count:%d\n",odd);

printf("Even numbers count:%d",even);

return 0;

}

1. Write a C program to check if array contains a duplicate number.

CODE:

#include <stdio.h>

int main() {

int n,i,j;

printf("Enter no of elements:");

scanf("%d",&n);

int arr[n];

int dup=0;

for(i=0;i<n;i++){

printf("Enter the elements in a array:");

scanf("%d",&arr[i]);

}

for(i=0;i<n;i++){

for(j=i+1;j<n;j++){

if(arr[i]==arr[j]){

dup=1;

printf("Duplicate found :%d",arr[i]);

return 0;

}

}

}

if(dup=0){

printf("No duplicate found!!");

}

return 0;

}

1. Write a C program to reverse array in place in C.

CODE:

#include <stdio.h>

int main() {

int n,i,j;

printf("Enter no of elements:");

scanf("%d",&n);

int arr[n];

for(i=0;i<n;i++){

printf("Enter array elements:");

scanf("%d",&arr[i]);

}

int start=0,end=n-1;

while(start < end){

temp=arr[start];

arr[start]=arr[end];

arr[end]=temp;

start=start+1;

end=end-1;

}

printf("Reversed Array:");

for(i=0;i<n;i++){

printf("%d\t",arr[i]);

}

return 0;

}

1. Given an array of size n and a number k, find all elements that appear more than n/k times.

CODE:

#include <stdio.h>

int main() {

int arr[100], n, k, count;

printf("Enter the size of the array: ");

scanf("%d", &n);

printf("Enter %d elements:\n", n);

for (int i = 0; i < n; i++) {

scanf("%d", &arr[i]);

}

printf("Enter the value of k: ");

scanf("%d", &k);

printf("Elements that appear more than n/k times are:\n");

for (int i = 0; i < n; i++) {

count = 1;

int already\_counted = 0;

for (int j = 0; j < i; j++) {

if (arr[i] == arr[j]) {

already\_counted = 1;

break;

}

}

if (already\_counted)

continue;

for (int j = i + 1; j < n; j++) {

if (arr[i] == arr[j])

count++;

}

if (count > n / k)

printf("%d\n", arr[i]);

}

return 0;

}

8) Write a program that accepts an array and a key value. Rotate the array element by ‘key’ times.

Example:

Input: array[]= [1, 2, 3, 4, 5, 6]

key=2

Output : [ 3, 4, 5, 6, 1, 2]

CODE:

#include <stdio.h>

int main() {

int n,i,j,key;

printf("Enter no of elements:");

scanf("%d",&n);

int arr[n],rotated[n];

for(i=0;i<n;i++){

printf("Enter array elements:");

scanf("%d",&arr[i]);

}

printf("Enter key value:");

scanf("%d",&key);

key=key%n;

for(i=0;i<n;i++){

rotated[i]=arr[(i+key)%n];

}

printf("Rotated Array:");

for(i=0;i<n;i++){

printf("%d\t",rotated[i]);

}

return 0;

}

9) Write a C program to merge sorted array?

CODE:

#include <stdio.h>

int main() {

int n;

printf("Enter no of elements:");

scanf("%d",&n);

int arr1[n];

int arr2[n];

printf("Enter Array1 Elements:\n");

for(int i=0;i<n;i++){

scanf("%d",&arr1[i]);

}

printf("Enter Array2 Elements:\n");

for(int i=0;i<n;i++){

scanf("%d",&arr2[i]);

}

int size1=sizeof(arr1)/sizeof(int);

int size2=sizeof(arr2)/sizeof(int);

int merge[size1+size2];

int i=0,j=0,k=0;

while(i<size1 && j<size2){

if(arr1[i]<arr2[j]){

merge[k]=arr1[i];

i++;

}

else{

merge[k]=arr2[j];

j++;

}

k++;

}

while(i<size1){

merge[k]=arr1[i];

i++;

k++;

}

while(j<size2){

merge[k]=arr2[j];

j++;

k++;

}

printf("Merge sort:\n");

for(int x=0;x<size1+size2;x++){

printf("%d\t",merge[x]);

}

return 0;

}

10) Write a C program to check if array contains a duplicate number.

11) Write a C program to reverse array in place in C.

12)Check if a given array contains duplicate elements within k distance from each other Given an unsorted array that may contain duplicates. Also given a number k which is smaller than size of array. Write a C function that returns true if array contains duplicates within k distance.

CODE:

#include <stdio.h>

int main() {

int arr[100], n, k;

int found = 0;

printf("Enter size of array: ");

scanf("%d", &n);

printf("Enter %d elements:\n", n);

for (int i = 0; i < n; i++) {

scanf("%d", &arr[i]);

}

printf("Enter value of k: ");

scanf("%d", &k);

for (int i = 0; i < n; i++) {

for (int j = i + 1; j <= i + k && j < n; j++) {

if (arr[i] == arr[j]) {

found = 1;

break;

}

}

if (found)

break;

}

if (found)

printf("Duplicates found within distance %d\n", k);

else

printf("No duplicates found within distance %d\n", k);

return 0;

}

**Deadline: 10.09.2025**