ASSIGNMENT – 15

1. Write a function to find the greatest number from the given array of any size. (TSRS)

#include<stdio.h>

int max\_arr(int a[],int size)

{

    int i,temp;

    temp=a[0];

    for(i=0;i<size-1;i++)

    {

        if(temp<a[i+1])

            temp=a[i+1];

    }

    return temp;

}

int main()

{

    int n,i,max;

    printf("Enter the no. of elements : ");

    scanf("%d",&n);

    int s[n];

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

    for(i=0;i<n;i++)

        scanf("%d",&s[i]);

    max=max\_arr(s,n);

    printf("Greatest element of the array is : %d",max);

    return 0;

}

2. Write a function to find the smallest number from the given array of any size. (TSRS)

#include<stdio.h>

int min\_arr(int a[],int size)

{

    int i,temp;

    temp=a[0];

    for(i=0;i<size-1;i++)

    {

        if(temp>a[i+1])

            temp=a[i+1];

    }

    return temp;

}

int main()

{

    int n,i,min;

    printf("Enter the no. of elements : ");

    scanf("%d",&n);

    int s[n];

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

    for(i=0;i<n;i++)

        scanf("%d",&s[i]);

    min=min\_arr(s,n);

    printf("Smallest element of the array is : %d",min);

    return 0;}

3. Write a function to sort an array of any size. (TSRS)

#include<stdio.h>

int sort\_arr(int a[],int size)

{

    int i,j,temp;

    for(i=0;i<size-1;i++)

    {

        for(j=i+1;j<size;j++)

        {

            if(a[i]>a[j])

            {

                temp=a[i];

                a[i]=a[j];

                a[j]=temp;

            }

        }

    }

}

int main()

{

    int n,i,sort;

    printf("Enter the no. of elements : ");

    scanf("%d",&n);

    int s[n];

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

    for(i=0;i<n;i++)

        scanf("%d",&s[i]);

    printf("Sorted Array is : ");

    sort\_arr(s,n);

    for(i=0;i<n;i++)

        printf("%d ",s[i]);

    return 0;

}

4. Write a function to rotate an array by n position in d direction. The d is an indicative value for left or right. (For example, if array of size 5 is [32, 29, 40, 12, 70]; n is 2 and d is left, then the resulting array after left rotation 2 times is [40, 12, 70, 32, 29] )

#include<stdio.h>

void print\_arr(int arr[],int n)

{

    int i;

    for(i=0;i<n;i++)

        printf("%d ",arr[i]);

}

void create\_arr(int arr[],int n)

{

    int i;

    for(i=0;i<n;i++)

        scanf("%d",&arr[i]);

}

int rotate\_arr(int arr[],int num,int n,int d)

{

    int i,j,temp;

    if(d==0)

    {

        for(i=0;i<n;i++)

        {

            temp=arr[0];

            for(j=1;j<num;j++)

            {

                arr[j-1]=arr[j];

            }

            arr[num-1]=temp;

        }

    }

    else if(d==1)

    {

        for(i=0;i<n;i++)

        {

            temp=arr[num-1];

            for(j=num-1;j>0;j--)

            {

                arr[j]=arr[j-1];

            }

            arr[0]=temp;

        }

    }

    else

    {

        printf("Wrong Input!!!");

        printf("Enter the direction again (0 for left and 1for right) : ");

        scanf("%d",&d);

        rotate\_arr(arr,num,n,d);

    }

}

int main()

{

    int num,n,d;

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

    scanf("%d",&num);

    int a[num];

    printf("Enter the elements of Array : ");

    create\_arr(a,num);

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

    scanf("%d",&n);

    printf("Enter the direction (0 for left and 1for right) : ");

    scanf("%d",&d);

    rotate\_arr(a,num,n,d);

    printf("Array after rotation is : ");

    print\_arr(a,num);

    return 0;

}

5. Write a function to find the first occurrence of adjacent duplicate values in the array. Function has to return the value of the element.

#include<stdio.h>

int copy\_arr(int arr[],int a[],int n)

{

    int i;

    for(i=0;i<n;i++)

        a[i]=arr[i];

}

void create\_arr(int arr[],int n)

{

    int i;

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

    for(i=0;i<n;i++)

        scanf("%d",&arr[i]);

}

int dupli\_element(int arr[],int n)

{

    int i;

    for(i=0;i<n-1;i++)

    {

        if(arr[i]==arr[i+1])

            return arr[i];

    }

    return(printf("Not found any adjacent duplicate element ."));

}

int main()

{

    int n;

    printf("Enter limit of array : ");

    scanf("%d",&n);

    int arr[n];

    create\_arr(arr,n);

    printf("First Adjacent duplicate element found is : %d",dupli\_element(arr,n));

    return 0;

}

6. Write a function in C to read n number of values in an array and display it in reverse order.

#include<stdio.h>

int reverse\_arr(int arr[],int n)

{

    int i;

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

    for(i=0;i<n;i++)

        scanf("%d",&arr[i]);

    printf("Displaying the array in reverse order : ");

    for(i=n-1;i>=0;i--)

        printf("%d ",arr[i]);

}

int main()

{

    int n;

    printf("Enter limit of array : ");

    scanf("%d",&n);

    int arr[n];

    reverse\_arr(arr,n);

    return 0;

}

7. Write a function in C to count a total number of duplicate elements in an array.

#include<stdio.h>

void create\_arr(int arr[],int n)

{

    int i;

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

    for(i=0;i<n;i++)

        scanf("%d",&arr[i]);

}

int dupli\_count(int arr[],int n)

{

    int i,j,count=0;

    for(i=0;i<n-1;i++)

    {

        for(j=i+1;j<n;j++)

        {

            if(arr[i]==arr[j])

            {

                count++;

                break;

            }

        }

    }

    return count;

}

int main()

{

    int n;

    printf("Enter limit of array : ");

    scanf("%d",&n);

    int arr[n];

    create\_arr(arr,n);

    printf("Total no. of duplicate elements in array are : %d",dupli\_count(arr,n));

    return 0;

}

8. Write a function in C to print all unique elements in an array.

#include<stdio.h>

int unique\_element(int arr[],int n,int max)

{

    int i,a[max+1];

    for(i=0;i<=max;i++)

        a[i]=0;

    for(i=0;i<n;i++)

        a[arr[i]]++;

    printf("Unique elements are : ");

    for(i=0;i<=max;i++)

    {

        if(a[i]==1)

            printf("%d ",i);

    }

}

int max\_arr(int a[],int size)

{

    int i,temp;

    temp=a[0];

    for(i=0;i<size-1;i++)

    {

        if(temp<a[i+1])

            temp=a[i+1];

    }

    return temp;

}

void create\_arr(int arr[],int n)

{

    int i;

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

    for(i=0;i<n;i++)

        scanf("%d",&arr[i]);

}

int main()

{

    int n,i,max;

    printf("Enter limit of array : ");

    scanf("%d",&n);

    int arr[n];

    create\_arr(arr,n);

    max=max\_arr(arr,n);

    unique\_element(arr,n,max);

    return 0;

}

9. Write a function in C to merge two arrays of the same size sorted in descending order.

//a function in C to merge two arrays of the same size sorted in descending order

#include<stdio.h>

void create\_arr(int arr[],int n)

{

    int i;

    for(i=0;i<n;i++)

        scanf("%d",&arr[i]);

}

int sort\_arr(int a[],int size)

{

    int i,j,temp;

    for(i=0;i<size-1;i++)

    {

        for(j=i+1;j<size;j++)

        {

            if(a[i]<a[j])

            {

                temp=a[i];

                a[i]=a[j];

                a[j]=temp;

            }

        }

    }

}

int merge\_arr(int arr1[],int arr2[],int n)

{

    int s[2\*n],i=0,j=0,k;

    for(k=0;k<2\*n;k++)

    {

        if(i>=n)

        {

            while(k<2\*n)

            {

                s[k]=arr2[j];

                j++;

                k++;

                if(k==10)

                    break;

            }

        }

        else if(j>=n)

        {

            while(k<2\*n)

            {

                s[k]=arr1[i];

                i++;

                k++;

                if(k==10)

                    break;

            }

        }

        else if(arr1[i]<arr2[j])

        {

            s[k]=arr2[j];

            j++;

        }

        else if(arr1[i]>arr2[j])

        {

            s[k]=arr1[i];

            i++;

        }

        else if(arr1[i]==arr2[j])

        {

            if(i>j)

            {

                s[k]=arr2[j];

                j++;

            }

            else

            {

                s[k]=arr1[i];

                i++;

            }

        }

    }

    for(i=0;i<2\*n;i++)

        printf("%d ",s[i]);

}

int main()

{

    int n,i,max;

    printf("Enter limit of array : ");

    scanf("%d",&n);

    int arr1[n],arr2[n];

    printf("Enter the elements of array 1 : ");

    create\_arr(arr1,n);

    printf("Enter the elements of array 2 : ");

    create\_arr(arr2,n);

    sort\_arr(arr1,n);

    sort\_arr(arr2,n);

    merge\_arr(arr1,arr2,n);

    return 0;

}

10. Write a function in C to count the frequency of each element of an array.

#include<stdio.h>

int frequency\_element(int arr[],int n,int max)

{

    int i,a[max+1];

    for(i=0;i<=max;i++)

        a[i]=0;

    for(i=0;i<n;i++)

        a[arr[i]]++;

    printf("Frequency of elements are : ");

    for(i=0;i<=max;i++)

    {

        if(a[i]!=0)

            printf("\n%d => %d",i,a[i]);

    }

}

int max\_arr(int a[],int size)

{

    int i,temp;

    temp=a[0];

    for(i=0;i<size-1;i++)

    {

        if(temp<a[i+1])

            temp=a[i+1];

    }

    return temp;

}

void create\_arr(int arr[],int n)

{

    int i;

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

    for(i=0;i<n;i++)

        scanf("%d",&arr[i]);

}

int main()

{

    int n,i,max;

    printf("Enter limit of array : ");

    scanf("%d",&n);

    int arr[n];

    create\_arr(arr,n);

    max=max\_arr(arr,n);

    frequency\_element(arr,n,max);

    return 0;

}